8. Problem Management Procedures

EMD Problem Management is administered through system-level and site-level control board reviews. These control boards oversee the analysis, recommendations, and actions taken to resolve ECS system/site problems concerning hardware, software, documentation, and procedures. Operations (OPS) Deployment and the site-level organizations resolve routine maintenance issues at the system-level and site-level, respectively, using the trouble ticket system for tracking system problems. Trouble Tickets (TT) may evolve into Non Conformance Reports (NCR), as required, which may then be utilized to generate Configuration Change Requests (CCR) to effect changes to the approved baseline. To ensure controlled change, NCRs are tracked using the DDTS in the EDF and CCRs are tracked manually by Systems Engineering.

The trouble ticket process is the first vehicle used to record and report problems with the operational system. Trouble Tickets can be generated by operations, maintenance, development, and customer personnel as well as external users. The trouble ticket process uses an automated database that tracks the resolution activities associated with each trouble ticket. Documentation that is related to the problem, and is not in electronic form, or is in electronic form at the DAAC, is distributed by the local CM Administrator (CMA), and is listed as an attachment to the trouble ticket.

The CM Administrator at each site serves as trouble ticket system administrator. Trouble Tickets initially generated at a site, the resolution of which require changes to the system level baseline, are forwarded to the SMC, where they are reviewed and *translated* into NCRs. Additionally, Trouble Tickets and CCRs that are generated at the sites, which are repaired locally, and result in site-unique extensions to the system level baseline, are forwarded to the SMC for tracking across the ECS/EMD baseline. The Problem Review Board (PRB) designee is responsible for tracking TTs after they have been received from the sites, and for propagating system problem resolutions for site visibility.

CMAs also support the activities of the local Problem Review Board. This includes generating status reports, and implementing resolutions, instructions, and changes as directed by the Board. User Services Representatives monitor trouble tickets to notify users concerning problem resolution and status. Maintenance engineers at respective levels will record all activities in the trouble ticket. This information can be used to determine critical maintenance concerns related to frequency of occurrence, criticality level, and the volume of problems experienced. The maintainability analysis will guide critical changes, volume and type of support components to be utilized, and will focus further development efforts.

This section provides an overview of the Trouble Ticketing process and defines the procedures for processing and resolving trouble ticket submissions. In addition, this section provides instructions for diagnosing network problems.

8.1 The Problem Resolution Process — An Overview

8.1.1 ECS Internal Process

Any ECS user may submit Trouble Tickets using the local trouble ticketing system at any site. TT submission triggers an internal review by the site's review board. Primary objectives of the internal review are to quickly identify and correct problems that fall within the site's capability to maintain, review and validate the priority of the problem, and to elevate to the system level those problems that either exceed their capability to repair, or that require a change to the system level baseline.

Problems passed from the sites to the ECS system level, are passed by transferring the trouble ticket from the local trouble ticket system to the SMC trouble ticket system. Here they are reviewed by the EMD Problem Review Board (PRB), which hosts daily teleconferences, known as the PRB Telecon.

The permanent membership of the Problem Review Board is as follows:

a. Chair: Problem Management Lead or designee

b. Each DAAC: one member representative

c. Quality Assurance: one member representative

d. ESDIS M&O: one member representative

The roles and responsibilities of the participants in the PRB Telecon are to:

a) Follow a nominal agenda that includes the following discussions:

i) New trouble tickets.

ii) Deferred trouble tickets.

iii) Aging high trouble tickets.

iv) Review of all Severity 1 (Sev 1) NCRs (OPS and RelB).

v) Review of new OPS NCRs.

vi) Review NCRs in the Verified State of DDTS.

b) Review severity of each NCR according to the guidelines specified in the

Operations Class NCR Management Process (MO-1-003-5)

c) Convert any TTs that identify a system non-conformance and have the appropriate information will be forwarded into an NCR by requesting that the Problem Management Lead forward the TT to the DDTS (Distributed Defect Tracking System). NCRs document the system non-conformance and are worked by the Sustaining Engineering staff.

d) NCRs that are deemed to not have enough information for proper adjudication will either be rejected as incomplete or they will be deferred until a future meeting as determined by the

PRB chairperson. The PRB administrator will maintain a list of deferred items and place those items on a future PRB agenda after more information is gathered.

The PRB performs a preliminary review of each trouble ticket to confirm the severity assigned by the site, the completeness of information and data relevant to the problem, and whether it requires a change to the system level operational baseline.

The PRB may forward TTs to the appropriate organizations or individuals for further analysis and technical investigation, resolution proposal, and NCR preparation, when required. The CM Administrator or DAAC Support Help Desk members provide administrative support to the PRB by publishing minutes of teleconference meetings, tracking TTs and action items, and by translating TTs to NCRs.

The PRB has the authority to direct resolutions to trouble ticket problems that do not change, or in any way affect, the ECS/EMD operational baseline and baseline documentation. An NCR is required when the Technical Investigation (TI) determines that the operational baseline must be changed in order to correct the problem identified in the trouble ticket.

The PRB is not a voting board; the membership is appointed for the purpose of providing timely, direct technical support to the Chair, who has the decision-making responsibility and authority.

The Science Development (SCDV) CCB is not a voting board. It does have the authority and responsibility to approve Class II changes to the operational baseline. (Class II changes are described in the Configuration Management Plan for the EMD Project (110-EMD-001).) Specific responsibilities include:

a. Review, approve and schedule; review and backlog; or reject each NCR's proposed resolution, or cost and schedule input from the Responsible Engineer (RE).

b. Approve the schedule for the deployment of configuration changes in the form of a 'drop' to the SMC.

c. Approve the content of each block.

d. Manage and adjust the schedule and contents of each block in accordance with program priorities and the progress of NCR work-off.

e. Review the status of all backlogged NCRs on a periodic basis. Schedule NCRs for a future block as appropriate

f. Collect and report on NCR statistics.

The permanent membership of the SCDV CCB is as follows:

a. Chair: EMD Systems Engineering and Integration Team (SEIT) designee.

b. Members of the EMD SEIT and designees.

c. Each DAAC: one representative.

8.2 Problem Management Procedures

The trouble ticket system is comprised of the Remedy Action Request System, a Commercial Off-The-Shelf (COTS) product that provides a distributed trouble ticketing service which provides a common environment and means of classifying, tracking, and reporting problem occurrence and resolution to both ECS users and operations personnel. The trouble ticketing service:

- Provides a GUI for operations personnel to access all Trouble Ticket services.
- Provides a common Trouble Ticket entry format.
- Stores Trouble Tickets.
- Retrieves Trouble Tickets via ad hoc queries.
- Allows operations personnel to forward problems from one DAAC to another.
- Produces stock and common reports.
- Provides an interface to user's and operator's e-mail to provide automatic notification.
- Offers an application programming interface through which applications can submit trouble tickets.
- Provides summary information to the SMC from each DAAC to allow trend reports regarding trouble tickets.
- Defines a consistent "life cycle" for trouble tickets.
- Allows each DAAC a degree of customization through definition of further re-prioritization and action rules.

In addition to the functionality provided by Remedy's Action Request System, the trouble ticketing service utilizes a set of custom HTML pages ("screens") to provide registered users with the ability to submit new trouble tickets and query the current status of any of their previous entries. Access to the Trouble Ticketing System through this technique provides users an easy method for reporting problems in an environment with which most are already familiar. Additionally, as another means of trouble ticket entry, the trouble ticket system provides a text e-mail template through which automated entry of trouble tickets is possible. Support staff members are able to enter Trouble Tickets through the trouble ticket system interface for problems received via other methods (for example, phone calls).

The Remedy Action Request System also functions as the User Contact Log. Remedy's Action Request System is configured to have a separate form that contains the entries that User Services personnel enter for each contact that they receive from a user. The User Contact Log allows a trouble ticket to be initiated from a log entry: with the push of a button — the Trouble Ticket will be populated with information from the contact log.

Users submit trouble tickets to the User Services Desk. External users submit trouble tickets through the Internet [using a series of hypertext mark-up language (HTML) screens]. Site personnel submit trouble tickets via the trouble ticket system (Remedy).

User Services personnel process trouble tickets for problem resolution in accordance with local policy. Trouble tickets are first evaluated to determine the severity of the problem and assignment of on-site responsibility. Every trouble ticket is logged into the database for record keeping purposes. Trouble tickets that can be resolved locally are assigned and tracked at the local center. The Operations Supervisor reviews each trouble ticket (for priority verification and problem description) and assigns it to an appropriate Maintenance Engineer for resolution.

Matters that require external or higher level assistance and problems, the repair of which require changes to the system baseline; are escalated via the PRB Telecon for discussion and disposition. The telecon is held to coordinate trouble ticket activities within the EMD organization as well as with development, customer, and user organizations.

8.3 Using the Trouble Ticket System

1. User or Operator discovers a problem with ECS (hardware, software, documentation, procedure) and documents this problem for later resolution. The submitter forwards a trouble ticket to User Services by: calling up the trouble ticket system via the Internet; phoning User Services; or sending an e-mail message to the trouble ticket system.
2. The trouble ticket is logged into the system. It automatically assigns "New" status to the trouble ticket and notifies the Operations Supervisor for assignment and prioritization. The status of each trouble ticket is the responsibility of Operations as it progresses through the resolution process.
3. Categorizing Problem Severity. The DAAC Help Desk will review the information provided by the DAAC to determine if the problem has been described in enough detail to warrant the recommended severity. If there is insufficient information, the DAAC submitter or point of contact will be contacted. The Trouble Ticket will be deferred to the Problem Review Board for further review.

In determining severity of the problem, the DAAC Help Desk must consider the following factors:

- Impact on the ability to ingest, process or distribute satellite data
- Frequency of occurrence
- Availability of an adequate work-around

Refer to Figure 8.3-1 to determine problem severity.

As Documented in NASA 420-05-03	As Used/Interpreted by M&O
Category 1: System/Service cannot perform critical function or imposes major safety hazard. (Priority 1) Presents an immediate impact to development, operations, services, or data processing functions; imposes major safety hazard to personnel, systems, or space mission resources; or results in loss of one or more essential mission objectives.	**HIGH (Severity 1)**: An NCR which causes: – Inability to perform a mission-critical function (i.e., Ingest/Pre-Processing/Archiving of Science Data, Planned Processing, Browse/Order/Distribute); – Performance of a mission-critical function to be so degraded that production minimum goals cannot be achieved; – A mission-critical function to be performed improperly, resulting in permanent loss of data; ***and*** for which no workaround exists ***or*** for which no workaround can be accommodated by DAAC operators given a detailed workaround procedure is documented but the procedure is inadequate based upon the complexity of the procedure, the abilities of an adequately trained and experienced operator, or both.
Category 2: System/Service substantially impaired. (Priority 2) Substantially impacts development, operations, services, or data processing functions; fails to operate within critical performance specifications; or cannot effectively or efficiently fulfill baseline requirements.	**MEDIUM (Severity 2)**: An NCR with the consequence that: – The performance of a mission-critical function is degraded and may prevent achieving production minimum goals; – A mission-critical function can be only partially performed, or performs improperly, resulting in temporary loss of data or incorrect data results; – A situation (actually or potentially) severly compromises ECS mission readiness or operational integrity; – A condition exists to produce a severely degraded mission-critical function, but a workaround will allow operations to continue temporarily without permanent loss of data or severely impaired performance/workload/schedules.
Category 3: System/Service slightly impaired. (Priority 3) Causes minor or no substantial impact to development, operations, services, or data processing functions. Support may be degraded, but mission can still be accomplished.	**Severity 3**: An NCR with the consequence that: – A non-critical mission function (e.g., Advertising) cannot be performed, or yields incorrect results; – Unexpected events occur which can be corrected using normal operational procedures with minimal impacts to performance/workloads/schedules – A condition exists to produce a degraded mission-critical function, but a workaround will allow operations to continue indefinitely without severely impaired performance/workload/schedules.
	Severity 4: Improvement (Nuisance; e.g., a typo).
	Severity 5: Enhancement (Identified for next release).

Figure 8.3-1. Trouble Ticket Priority / NCR Severity

4. All affected Operations Supervisors at the sites (SMC, DAACs, EOC, EDF) are notified by e-mail of the problem and solicited for inputs to problem assessment (impact) and resolution.
5. The Trouble Ticket database is updated by the CM Administrator or the DAAC Support Help Desk Staff. The trouble ticket may be modified to reflect any new information/coordination activity.
6. The Operations Supervisor assigns the problem to a Problem Investigator for further follow-up.
7. The Problem Investigator coordinates input from developers, vendors, and external organizations to effect the local resolution. The Problem Investigator presents significant issues at the PRB Telecon.
8. The Problem Investigator updates the trouble ticket database.
9. The Problem Investigator forwards any information regarding proposed/implemented fixes to the established notification list.
10. In those cases where the problem resolution will result in a change to the operational baseline, or where the local site wishes to elevate the TT to the PRB for advice or for resolution, the site CMA forwards the TT to the SMC.

11. The proposed resolution is then presented to the Problem Review Board for review, ratification, or revision.
12. Changes that do not affect controlled configuration items may be approved and implemented by the Problem Review Board and closed.
13. All system level changes are proposed in Non Conformance Reports (NCRs). The SMC opens an NCR in the Change Request Manager and closes the trouble ticket in Remedy. Emergency fixes (High TTs) can be made locally with the approval of the local CCB and then be forwarded to the SMC in a trouble ticket.
14. The off-site problem resolution process is monitored by the EMD Problem Review Board, which may also revise the proposed solution because of any system-level effect(s).
15. The NCR may be escalated to higher levels for system and/or external elements that may be involved in the resolution process.

Table 8.3-1 provides the activity checklist for the Trouble Ticket System.

Table 8.3-1. Trouble Ticket System - Activity Checklist

Order	Role	Task	Section
1	ECS users	Access the Trouble Ticket System	8.3.1, 8.4.1
2	ECS users	Submit Trouble Ticket	8.3.2
3	Maintenance Engineer	Modify Open Trouble Ticket	8.3.3
4	Operations Supervisor, Maintenance Engineer	Forward Trouble Ticket	8.3.4
5	Database Administrator	Add Users to Trouble Ticket System	8.3.5
6	CMA	Modify Trouble Ticket System User Privileges	8.3.6
7	CMA	Modify Trouble Ticket System Configuration	8.3.7
8	Maintenance Engineer, CMA	Generate Reports	8.3.8
9	Maintenance Engineer	Maintain Escalation Time Table	8.3.9

8.3.1 Accessing the Trouble Ticket System

The trouble ticket system may be accessed through either HTML or Remedy. The Trouble Ticket HTML is used by both User Services and the end user to submit trouble tickets without going through Remedy. It is accessed through the web. Through HTML, the user can submit, obtain a list, and view details of trouble tickets. Complete and detailed instructions for Remedy may be found in the current DID 609 (e.g., 609-EMD-001) and *Remedy's Action Request System Users's guide.*

Through Remedy, the User clicks on the User Tool icon, which opens the form to submit, query, or work a Trouble Ticket. The Main Remedy Trouble Ticket screen is used to select the appropriate forms for submitting, modifying, or displaying a trouble ticket. The Main Page data fields are identified in Table 8.3.1-1.

The Remedy Action Request System provides a distributed trouble ticketing service that furnishes DAACs a common environment and the means of classifying, tracking, and reporting problem occurrences and resolutions to both ECS users and operations personnel.

Several timesaving features are available through Remedy: the Admin Tool, Remedy Import tool, the Hardware Information form, and the GUI Notification tool. Brief descriptions are provided in Sections 8.3.1.1 through 8.3.1.4.

Table 8.3.1-1. RelB-Trouble Ticket Field Description (1 of 2)

Field Name	Data Type	Size	Entry	Description
Ticket-Id	Character	15	System generated	Ticket number which is set and maintained by the system
Ticket Status	Selection	4	Required	Status of the Trouble Ticket
Assigned-Priority	Selection	4	Required	Priority of Trouble Ticket assigned at the site (HIGH, MEDIUM, LOW)
Short Description	Character	128	Required	Short Description of the problem
Submitter Impact	Selection	4	Optional	Impact of the problem to the submitter (HIGH, MEDIUM, LOW)
Long-Description	Character	4060	Optional	Long Description of the problem
Resolution Log (End User Sees)	Diary	Unlim	Optional	General steps in the resolution of the problem
Detailed Resolution Log	Diary	Unlim	Optional	Detailed steps in the resolution of the problem
Submitter ID	Character	30	Required	User Id of the Submitter.
Submitter Name	Character	30	Optional	Full Name of the Submitter
Submitter Phone	Character	30	Optional	Phone number of the Submitter
Submitter e-mail	Character	64	Optional	E-mail address of the Submitter
Submitter Home DAAC	Character	60	Optional	Home DAAC of the Submitter
History	Diary	Unlim	Optional	Upon submission or modification, the person assigned to the ticket and the ticket status will be indicated in the History field. Due to a limitation in Remedy, this information will only be written when the Assigned-to and Status fields are modified
CI	Character	30	Required	Name of the Configuration Item to which the problem is associated
Assigned-To	Character	30	Optional	Person that Trouble Ticket has been assigned to

Table 8.3.1-1. RelB-Trouble Ticket Field Description (2 of 2)

Field Name	Data Type	Size	Entry	Description
Last-modified-by	Character	30	System generated	Person that last modified the Trouble Ticket
Create-date	Date/Time	4	System generated	Date Trouble Ticket was created at the present site
Last-Modified-date	Date/Time	4	System generated	Date the Trouble Ticket was last modified
Related CCR	Character	60	Optional	ID of a related CCR
Key Words	Character	255	Optional	Key words to help identify this Trouble Ticket
Problem Type	Character	30	Required	Type of problem addressed by this Trouble Ticket
Closing Code	Character	60	Required	Disposition of the closed trouble ticket
Closed-by	Character	60	System generated	Person that closed this Trouble Ticket
Close-date	Date/Time	4	System generated	Date this Trouble Ticket was closed
Software Resource	Character	60	Optional	Software Resource that the problem came from
Hardware Resource	Character	60	Optional	Hardware Resource that this problem came from
Duplicate Master Id	Character	25	Optional	The Master Ticket-ID of this Trouble Ticket
Forward-to	Character	60	Optional	Site that this Trouble Ticket was last forwarded to
Forwarded-from	Character	60	Optional	Site that forwarded this Trouble Ticket
Forwarded-by	Character	60	Optional	Contact person at the forwarding site
Forward-date	Date/Time	4	Optional	Date Trouble Ticket was forwarded
Unique-Identifier	Character	20	Optional	Unique identifier which is established at the origination site This identifier should NEVER be changed once set
Forwarded-to-1	Character	60	Optional	First site to have been forwarded this Trouble Ticket
Forwarded-to-2	Character	60	Optional	Second site to have been forwarded this Trouble Ticket
Forwarded-to-3	Character	60	Optional	Third site to have been forwarded this Trouble Ticket
Forwarded-to-4	Character	60	Optional	Fourth site to have been forwarded this Trouble Ticket
Associated Contact Log Id	Character	30	Optional	ID number of the Associated Contact Log

8.3.1.1 Remedy's GUI Admin Tool

Most Remedy administrative functions are accomplished using the Remedy Administration tool, implemented in the Windows environment on a personal computer. The log-in dialog provides an **Accounts . . .** button for selecting the user account and servers to be addressed in the administration session. When the account and servers have been selected and the log in is complete, the Remedy Administrator Server Window is displayed. In this window, the server directory structure may be expanded and an object type may be selected for performing administrative functions on various "objects" such as forms and groups (see Sections 8.3.5, 8.3.6, and 8.3.7).

8.3.1.2 Remedy Import Tool

The Remedy Import tool is used to import existing entries rather than retyping information manually. It also enables the user to import entries into a form from a file generated by the Admin tool. For more information on the Import tool, refer to the *Action Request System 4.5 Concepts Guide* and the *Action Request System 4.5 Workflow Administrator's Guide*.

8.3.1.3 Remedy's Hardware Information Form

Detailed hardware information can be provided beyond what can be entered on the Trouble Tickets form by using the Hardware Information form. The User Tools Hardware Information form provides the vehicle to add a description of a hardware problem that corresponds to a trouble ticket. Through this form, the user can enter detailed information about failed hardware components (e.g., part and serial numbers) and the actions taken to correct the problem. This form is accessed by opening RelB-Hardware Information form or via a Hardware Information link from the Trouble Tickets form.

8.3.1.4 Remedy's GUI Notification Tool

The GUI Notification Tool is used as an alternative to email notification to notify the user of a Remedy event. It allows properties and options to be modified via pull-down menus. Examples of GUI notification include a beep, a pop-up window, and a flashing message. In addition, both an email and a GUI notification can be sent if the site so desires.

8.3.2 Submitting a Trouble Ticket

When a problem is either found by or reported to User Services, follow the procedure applicable to your system, to create and log trouble tickets. Trouble tickets can be submitted via HTML or via Remedy's user tool – RelB-Trouble Tickets form. Remedy's Contact Log form is used to classify, track, and report contacts of ECS users and operators and also to submit a trouble ticket from a log entry. E-mail is another method of submitting a trouble ticket. The template is available from your System Administrator.

1. For HTML submission:

   a) Access HTML Trouble Ticketing Main page.

   b) Select Submit link, which opens the Submit page.

   c) Fill out the impact, short description, and detailed description fields.

   d) Select Submit.

2. For submission through Remedy:

   a) Access Remedy User Tool by logging in to the Remedy host and executing the command **aruser &**.

   b) Access the RelB-Trouble Tickets form by selecting **Open** from the **File** menu, highlighting **RelB_TroubleTickets**, and clicking the **New** button.

   c) Fill in those fields as specified in Table 8.3.1-1 "RelB-Trouble Ticket Field Description".

   d) Select **Save** from the **Actions** menu (or click on the **Save** button).

3. For submission from a Remedy Contact Log entry:

   a) Open RelB-Contact Log form.

   b) Fill out Contact Log ID and Contact Information. If the contact is a registered Remedy user, the contact information is filled out automatically.

   c) Fill in Short Description (limit is 128 characters).

   d) Click on **Create TT** button.

4. For submission via E-mail:

   a) Obtain Template from your System Administrator.

   b) Address the message to arsystem@____________.___________.________.

   c) Copy template into message area. DO NOT INCLUDE AS AN ATTACHMENT. DO NOT ALTER TEMPLATE. The template is presented in Figure 8.3.2-1. The # sign indicates comments, which are not read by Remedy. **Enter data as indicated in Figure 8.3.2-1.** Send message.

```
#
#  File exported Wed Feb 28 19:01:27 1996
#
Schema: RelB-Trouble Tickets
Server: remedy server name
Login:
Password:                                         Field ID internal
                                                  to Remedy

       Short Description !        8!:                          Default
        Submitter Impact !536870922!: Low                      value
# Values: Low, Medium, High                                    Select one
        Long-Description !        9!:
            Submitter ID !        2!:
          Submitter Name !536870917!:                    Enter data after
         Submitter Phone !536870918!:                    colon
         Submitter e-mail !536870921!:
     Submitter Home DAAC !536870919!:
```

Figure 8.3.2-1. Trouble Ticket E-mail Template

8.3.3 Reviewing and Modifying Open Trouble Tickets

Trouble tickets may need to be modified based on better understanding of the nature of problems defined and revised resolutions from the Maintenance Engineer investigations, Sustaining Engineering inputs, Developer inputs, Problem Review Board decisions, Change Control Board decisions, and/or Failure Review Board decisions. The results will be factored into revisions and/ or additions to the Trouble Ticket log.

1. For HTML Review and Modification of Trouble Tickets:

   a) Access HTML Trouble Ticketing Main (see Section 8.4). Trouble Tickets can be ***submitted, queried or modified***.

   b) Select List link, which opens the List page and shows each Trouble Ticket's Identification, Short Description, and Status.

   c) Select the Trouble Ticket Id to get a more detailed description of that particular Trouble Ticket.

2. For Reviewing and Modifying Trouble Tickets through Remedy:

   a) Access Remedy User Tool (see Section 8.3.2.2a).

   b) Access the RelB-Trouble Tickets form by selecting **Open** from the **File** menu, highlighting **RelB_TroubleTickets**, and clicking the **Search** button.

c) In the resulting **Modify RelB-TroubleTickets** window, enter a search criterion and then select **Search** from the **Actions** menu (or execute that menu selection with no search criterion entered, to obtain a list of all Trouble Tickets, and then select one to modify).

d) Review and/or modify the Trouble Ticket fields as necessary.

e) Select **Save** from the **Actions** menu (or click on the **Save** button).

8.3.4 Forwarding Trouble Tickets

Trouble ticket administrative reports are forwarded for local and system-wide usage. The trouble ticket contains all forwarding information; once forwarded, it goes to the RelB-TT-ForwardToSite holding area (transparent to the user). The RelB-TT-Sites schema is used to indicate the site name and email address to be used in forwarding. To forward a trouble ticket:

1. Access the Trouble Ticket to be forwarded (see 8.3.3a to 8.3.3c).
2. Select **Forwarded** from the pull-down menu in the **Ticket Status** field.
3. Select the TT's destination using the **Forward-to** field pick list.
4. Select the **Forward** button.
5. Select **Save** from the **Actions** menu (or click on the **Save** button).

8.3.5 Adding Users to Remedy

The TT Administrator uses the Remedy User form to grant access to the Remedy tool. Users who leave the EMD program can be deleted. The Remedy *Action Request System 4.5 Concepts Guide*, Chapter 6, "Controlling Access to AR System," summarizes access control and license elements. There are three classes of licenses that can be assigned to users:

- The **Read** class allows users to submit requests (e.g., Trouble Tickets, User Contact Log records) and modify their own requests. Thus, there are no license restrictions on the number of users who can be granted permission to create and read trouble tickets.
- The **Fixed Write** class provides the capabilities of the Read class plus the ability to modify existing requests submitted by others. A Fixed Write license assigned to a user is always reserved for that user. The AR system administrator must have a Fixed Write license.
- The **Floating Write** class provides the capabilities of the Fixed Write class, but these licenses are not reserved to a single user; instead, they are available on a first-come, first-served basis. Each DAAC has five Floating Write licenses. It is appropriate to assign Floating Write licenses to those who will need to modify Trouble Tickets (e.g., those who will be assigned to work and resolve problems). The limit of five licenses should pose no problem, as it is unlikely that there will be many instances in which more than five people will be trying to modify Trouble Tickets at one time.

The Remedy *Action Request System 4.5 Workflow Administrator's Guide*, Chapter 3, "Defining Access Control," provides information on using the **User** form (a "form" represents a table in the Remedy database) to add registered users.

8.3.6 Changing Privileges in Remedy

Changing privileges in Remedy, or controlling privileges of those who have access to Remedy, is done by the CM Administrator. There are numerous Remedy privilege groups for ECS, and a change to the privileges of any group requires an approved Configuration Change Request (CCR). Access privileges determine which forms a user may open and specify whether a user may change information in a field or merely view it.

The Remedy *Action Request System 4.5 Workflow Administrator's Guide*, Chapter 3, "Defining Access Control," provides detailed information about access control and privileges. The chapter includes a section on "Adding Groups" that provides instruction on using the **Group** form to create access control groups to which users may then be assigned. A user's privileges may be changed in two ways:

- Changing the group to which the user is assigned.
- Changing the access privileges of the group.

The **User** form is used to implement group assignment. To change the access privileges of a group, you use the **Admin** tool. There are two primary ways to change the access privileges of a group. Both ways use the **Group Permissions** window. This window is obtained by selecting **Groups** on the **Remedy Administrator – Server Window** and then double clicking on one of the listed groups. One way to change access privileges is to use the **Form Permissions** tab of the **Group Permissions** window to select which forms are available to members of the group when they access the **Open** dialog box of the **User** form. The second way is to use the **Field Permissions** tab to specify which fields of a form the members of the group may change and which fields they may view but not change. Groups have already been created to accommodate all privileges needed by Remedy Users for Release B. These groups are identified in Table 8.3.6-1.

Table 8.3.6-1. Table of Access Control Groupings (1 of 2)

Groups	Description	Access Type
Operator	Submits trouble ticket internally.	Change
User Services	Submits trouble ticket internally for user.	Change
Operations Supervisor	Assigns problem priority and resolution responsibility. Can forward trouble ticket to another site.	Change
Resource Manager	Assigns problem priority and resolution responsibility. Can forward trouble ticket to another site.	Change
Resolution Technician	Attempts to resolve problem.	Change

Table 8.3.6-1. Table of Access Control Groupings (2 of 2)

Groups	Description	Access Type
Problem Review Board Chair Person	Reviews proposed solutions.	Change
Administrator	Adds groups and users. Changes permissions. Sets escalation times. Sets menu items. Etc.	Change
Sub-Administrator	Same functions as Administrator but only with certain Schemas.	Change
Browser	Read only permission.	Read
Customize	Can use all features of the customize facility.	Change
Submitter	Place holder for anyone that submits a trouble ticket.	NA
Assignee	Place holder for anyone that is assigned a trouble ticket.	NA
Public	Read only permission. Guest users are automatically put in this group.	Read
NotifyNewEscal	Everyone that will be notified on an escalation due to trouble ticket being in “New” status.	Read
NotifyAssignedEscal	Everyone that will be notified on an escalation due to trouble ticket being in “Assigned” status.	Read
NotifySolPropEscal	Everyone that will be notified on an escalation due to TT being in “Solution Proposed” status.	Read
NotifyImpSolEscal	Everyone that will be notified on an escalation due to trouble ticket being in “Implement Solution” status.	Read
NotifySolImpEscal	Everyone that will be notified on an escalation due to TT being in “Solution Implemented” status.	Read

8.3.7 Modifying Remedy’s Configuration

RelB-Trouble Ticket forms’ pulldown menus can be customized. Customization is achieved through the Admin Tool by modifying the RelB-Menu-Closing Codes, RelB-Menu-Hardware Resources, RelB-Menu-Software Resources, RelB-Menu-Key Words, RelB-Menu-Problem Type, or Sites forms.

1. To modify the Remedy environmental variables, refer to the *Action Request System 4.5 Workflow Administrator’s Guide*.

   NOTE: No administrative configuration should be made without proper configuration change approval.

8.3.8 Generating Trouble Ticket Reports

A set of predefined reports is available through Remedy. These reports are trouble ticket administrative reports generated for local and system-wide usage. There are several types of predefined reports, including:

- Assigned-to Report – provides a report of the number of Tickets assigned to technicians.
- Average Time to Close TTs – provides a report of the average time to close trouble tickets.
- Hardware Resource Report – provides a report sorted and grouped by Hardware Resources and Closing Codes.
- Number of Tickets by Status – provides the number of Trouble Tickets grouped by Status.
- Number of Tickets by Priority – provides the number of Trouble Tickets grouped by assigned priority.
- Review Board Report – provides a report of the details of TTs for the TT Review Board.
- SMC TT Report – provides a report to be sent to the SMC.
- Software Resource Report – provides a report sorted by Software Resources and their Closing Codes.
- Submitter Report – indicates by submitter the number and type of trouble tickets in the system.
- Ticket Status Report – provides a report sorted and grouped by Ticket Status.
- Ticket Status by Assigned-to – provides a report sorted and grouped by the last person assigned to a Trouble Ticket.

Most of the time, you will probably select a report from the list, using the **Report** window. If you need a custom report, it is possible to select data from the database and create a report using an appropriate separate software application (e.g., a spreadsheet program).

8.4 Using Hypertext Mark-up Language (HTML) Screens

The hypertext mark-up language (HTML) Trouble Ticket Main Screen ("ECS Trouble Ticketing: Menu") provides an introduction on how to use the Trouble Ticketing HTML, and is used by registered users to go to either the Submit page or List page.

Selecting **<u>Submit a Trouble Ticket</u>** will bring up the Trouble Ticketing Submit screen.

Selecting **<u>List the [username] Trouble Tickets</u>** will bring up the Trouble Ticketing List screen.

Help on the Trouble Ticket HTML screens is available by clicking on the Trouble Ticket Help icon at the bottom of the screen.

8.4.1 ECS Trouble Ticketing HTML Submit Screen

The HTML Trouble Ticket Submit screen is used by registered users to submit a Trouble Ticket.

Table 8.4.1-1 below provides a description of the Trouble Ticket HTML Submit Screen fields.

Table 8.4.1-1. Trouble Ticket HTML Submit Screen Field Description

Field Name	Data Type	Size	Entry	Description
ID	character	30	System generated	Submitter Id
Name	character	30	System generated	Submitter Name
E-mail address	character	64	System generated	Submitter E-mail Address
Phone	character	30	System generated	Submitter Phone Number
Home DAAC	character	60	System generated	Submitter Home DAAC
Impact	selection	4	Required	Impact to Submitter
Short description	character	125	Required	Short description of problem
Detailed problem description	character	245	Optional	Long description of problem

When the information is completed, the user can submit the Trouble Ticket by clicking on the **Submit** button on the lower half of the screen. The Problem Information Fields can be cleared by clicking on the **Reset** button. The user also has the choice of returning to the Trouble Ticketing Homepage or going to the Trouble Ticket Help screen by clicking on the respective icons at the bottom of the page.

8.4.2 ECS Trouble Ticketing HTML Success Screen

The HTML Trouble Ticket Success screen is used by registered users to ensure successful submission and report Trouble Ticket Id.

From this screen, the user is provided with the following information/options:

- Confirmation that the trouble ticket was successfully submitted, the trouble ticket identification number, and who submitted the trouble ticket.
- Notification that an E-mail message has been sent to the user indicating that a Trouble Ticket has been submitted and when it was closed. Selecting this Trouble Ticket will open the Trouble Ticket Detailed Screen.
- Instructions telling the user how to check the progress of Trouble Ticket resolution.

The user also has the choice of returning to the Trouble Ticketing Homepage or going to the Trouble Ticket Help screen by clicking on the respective icons at the bottom of the page.

8.4.3 ECS Trouble Ticketing HTML List Screen

The HTML Trouble Ticket List screen is used by registered users to List Trouble Tickets for a user and links the listed Trouble Ticket Number to the Trouble Ticket Detailed Screen.

Table 8.4.3-1 below provides a description of the Trouble Ticket HTML List Screen fields.

Table 8.4.3-1. Trouble Ticket HTML List Screen Field Description

Field Name	Data Type	Size	Entry	Description
Trouble Ticket Number	character	15	System generated	Trouble Ticket Id
Problem Short Description	character	125	System generated	Short Description of Problem
Status	character	20	System generated	Status of Trouble Ticket

The user also has the choice of returning to the Trouble Ticketing Homepage or going to the Trouble Ticket Help screen by clicking on the respective icons at the bottom of the page.

8.4.4 ECS Trouble Ticketing HTML Detailed Screen

The HTML Trouble Ticket Detailed screen is used by registered users to see a more detailed output of a Trouble Ticket.

Table 8.4.4-1 below provides a description of the Trouble Ticket HTML Detailed Screen fields.

Table 8.4.4-1. Trouble Ticket HTML Detailed Screen Field Description (1 of 2)

Field Name	Data Type	Size	Entry	Description
ID	character	30	System generated	Submitter Id
Name	character	30	System generated	Submitter Name
E-mail address	character	64	System generated	Submitter E-mail Address
Phone	character	30	System generated	Submitter Phone Number
Home DAAC	character	60	System generated	Submitter Home DAAC
Status	selection	4	System generated	Status of Trouble Ticket
Impact	selection	4	System generated	Impact to Submitter (low, medium, high)
Short description	character	125	System generated	Short description of problem

Table 8.4.4-1. Trouble Ticket HTML Detailed Screen Field Description (2 of 2)

Field Name	Data Type	Size	Entry	Description
Detailed problem description	character	245	System generated	Long description of problem
Log	character	unlim.	System generated	Diary of problem resolution

The user also has the choice of returning to the Trouble Ticketing Homepage or going to the Trouble Ticket Help screen by clicking on the respective icons at the bottom of the page.

8.4.5 ECS Trouble Ticketing HTML Help Screen

The HTML Trouble Ticket Help screen is used by registered users to get help with the HTML screens.

This screen provides general information on the following:

- Index -- links that scroll the screen to the Introduction, Submit Page, and List Page sections listed below.
- Introduction – provides information about the Trouble Ticket Help page.
- Menu Page – describes the Trouble Ticketing Menu page.
- Submit Page – describes the Trouble Ticket Submit page.
- Success Page – describes the Trouble Ticket Success page.
- List Page – describes the Trouble Ticket List page.
- Detailed Page - describes the Trouble Ticket Detailed page.

8.5 Emergency Fixes

Emergencies may be in real time with the understanding that the trouble ticket system must be brought up-to-date as soon as possible after implementing the repair. The example presented below, involves a hardware failure. The problem needs to be resolved quickly to bring a system back into operation. The resolution requires emergency replacement of a component that is of a later version than is contained in the original equipment. The scenario is summarized in Table 8.5-1.

Scenario — An Example of an Emergency Change Procedure

It is 7:00 on a Saturday evening. The DAAC operator detects a problem with the automated tape library (ATL) and reports the problem to the trouble ticket system. The trouble ticket is routed to the System Administrator, who confirms that the system will not operate and notifies the site Maintenance Engineer. After running further diagnostics, the Maintenance Engineer reports the problem and symptoms to the OEM’s maintenance desk. The original equipment manufacturer (OEM) maintenance representative arrives and concludes that a controller card has failed. The only card the OEM has immediately available is of a later version and no spares are available on site. It will be Monday at the earliest before a replacement board of the same revision level can be located. The site maintenance engineer reports this to the operations Crew Chief (i.e., shift leader) for a decision.

The DAAC cannot afford to have the ATL down until Monday. The Crew Chief calls the DAAC manager at home, apprises him of the situation, and obtains approval to replace the board with the later version if tests conclude that it works properly. The OEM's maintenance representative installs the board. The site's sustaining engineer tests the new controller board, finds that it works properly, and brings the ATL back on-line. The sustaining engineer updates the trouble ticket to document the configuration change and the authority for the change, and forwards it to the site CMA. The site Maintenance Engineer updates the property record with the model, version, and serial number of the new board.

The site CM Administrator reviews the trouble ticket, and presents it to the local CCB for approval. The CMA then updates the Baseline Manager with the new configuration and TT number authorizing the change. At this point, the site is operational at variance from the system baseline (i.e., site unique) and is at risk of losing maintenance support from OPS Deployment.

The site CMA forwards the trouble ticket to the SMC, presented to the PRB for priority review and is solved or translated into an NCR. The PRB reviews all emergency TTs to assess whether there may be impacts to the ECS and/or applicability to other sites. The SCDV CCB monitors all open NCR promotion and approves them for closure.

In the event that it is later discovered that the new version controller board has adverse impacts when operating in the ECS configuration, a board of the original version will have to be obtained to replace the newer version. In such cases, the action will be recorded on a new trouble ticket, citing the previous CCR.

Table 8.5-1 summarizes emergency procedures that might be taken during an after-hours, over-the-weekend emergency hardware failure.

Table 8.5-1. Example of Emergency Change Procedure (1 of 2)

Operator/User	System
Operator prepares trouble ticket to report ATL controller failure.	Trouble ticket recorded.
System Administrator and Maintenance Engineer confirm ATL controller failure, call ATL maintenance vendor, report call and time in Trouble ticket.	Diagnosis and vendor call recorded in trouble ticket.
Maintenance vendor isolates failure to the controller card. The later version card is the only card available.	
Crew Chief notified of situation and decision needed to bring ATL up to full operating capability. Approves use of the newer version card, records decision in the trouble ticket, forwards trouble ticket to Sustaining Engineer.	

Table 8.5-1. Example of Emergency Change Procedure (2 of 2)

Operator/User	System
Maintenance vendor installs card, tests using hardware diagnostics. Crew Chief authorizes controller to be brought back on-line.	
Maintenance Engineer records card installation by model/version into the trouble ticket.	Trouble ticket action recorded.
Sustaining Engineer reads trouble ticket and prepares for discussion at 8:30 am meeting. Updates the TT.	Install action recorded in TT. TT routed to the CMA.
CMA updates site baseline, forwards TT to the CCB. When CCB approves the action, CMA forwards to SMC.	Site ATL baseline updated in Baseline Manager.
PRB reviews emergency NCR, checks for applicability to other sites.	

This page intentionally left blank.

9. Configuration Management Procedures

The prepared procedures are applicable to all EMD hardware, software, and firmware components of systems or subsystems developed and acquired by the EMD contract and/or delegated to configuration management control by the operational site-level organizations. The procedures are applicable to all items maintained by the EMD Sustaining Engineering Organization in support of EMD mission-specific projects and multiple mission-specific institutional facilities. The procedures are not applicable to those entities controlled by higher level ESDIS Project Office CM Plans. CM procedures already in place may be used by the contractor subject to direction from the Change Control Board (CCB) chairperson.

Some major features of the approach being described here include:

- Customers participate in establishing the procedures;
- The SCDV CCB performs a support role for ESDIS and its designated on-site CCBs by processing system-level CCRs & Trouble Tickets;
- Prioritization, automated tools, and procedures are used for handling change requests;
- Diverse/Strategic representation at hierarchical CCBs facilitates a path for speedy escalation/resolution of problems/issues;
- Local organizations have the needed autonomy to accomplish their mission with the minimum necessary outside intervention to promote timely resolution of local problems and enable timely production of data products;
- Proper use and deployment of CM database assets to support all CCBs allows management monitoring, control, and analysis of activities;
- Coordination with the Failure Review Board allows coordinated response to problems and filtering of prioritized issues; and
- Common CM tools will be used in all elements of the EMD Project during operations.

The procedures are organized into seven major sections that address the flow-down of procedures from the EMD system-level to the site-level with references to site-tailoring of procedures where applicable. The topics include (Section 9.1) configuration identification, (Section 9.2) change control processes, (Section 9.3) configuration status accounting, (Section 9.4) configuration audits, (Sections 9.5 and 9.6) Software CM Manager (ClearCase), and (Section 9.7) Baseline Manager (ClearCase).

9.1 Configuration Identification Procedure

9.1.1 Purpose

The purpose of configuration identification during maintenance and operations is to incrementally establish and maintain the definitive basis of control and status accounting for the EMD control items. To accomplish configuration identification for both hardware and software, the configuration management (CM) administrator (CMA) shall ensure the maintenance of each EMD configuration controlled item in an operational baseline by executing the following tasks:

a. Assign identifiers to configuration items (CIs) and their component parts and associated configuration documentation, including revision and version number where appropriate. Assign serial and lot numbers, as necessary, to establish the CI effectivity of each configuration of each item of hardware and software.

b. Follow EMD developer guidelines as referenced below in Section 9.1.3.

c. Follow vendor nomenclature for COTS items.

d. Apply operation and maintenance (O&M) version name extensions to EMD modified item nomenclature following the rules in Section 9.1.4.

e. Follow author-designated version control and nomenclature for documents and follow guidelines from the EMD Librarian (cf. Chapter 20, *Library Administration*)

f. Support the EMD Librarian's efforts to maintain linkage of the EMD documentation to EMD configuration items in the Baseline Manager tool. Ensure that the marking and labeling of items and documentation with their applicable identifiers enables correlation between the item, configuration documentation, and other associated data.

g. Maintain a release system for configuration changes (cf. Section 9.2, *Configuration Change Control Procedures)*.

h. Maintain views of operational baselines using the Baseline Manager tool (cf. Section 9.7).

i. Ensure that applicable identifiers are embedded in the source and object code.

9.1.2 Applicability

All EMD CM Administrators and support personnel.

9.1.3 References

ESDIS CM Plan

Configuration Management Plan for the EMD Project	110-EMD-001
EMD Software Build Process	CM-1-045
Embedded Versioning	CM-1-031

Data Identification Numbering	DM-1-002
DoD Mil-Std-973	

9.2 Configuration Change Control Procedures

9.2.1 Purpose

The ESDIS CCB chartered Change Control Boards (CCBs) shall apply configuration control measures to all the EMD configuration items and the associated documentation prior to the time it is baselined for operations. The CCBs shall apply configuration control measures to

a. Ensure effective control of all CIs and their approved documentation;

b. Provide effective means, as applicable, for (1) proposing engineering changes to CIs, (2) requesting deviations and waivers pertaining to such items, (3) preparing notices of revision, and (4) preparing Specification Change Notices; and

c. Ensure the implementation of approved changes.

9.2.2 Applicability

All ESDIS chartered EMD CCBs.

9.2.3 References

ESDIS CM Plan	
EMD Configuration Management Plan	110-EMD-001
CCB Change Control Process	CM-1-004

9.2.4 Procedures

9.2.4.1 Configuration Change Request Preparation

The Configuration Change Request (CCR) form in Figure 9.2.4-1 has been developed as a medium for the drafting of CCRs throughout the EMD Operations Deployment environment for changes processed locally at EMD site-level chartered CCBs at the SMC, EOC, and DAACs (GSFC, LaRC, EDC, NSIDC, and ORNL). For changes processed by the ESDIS Change Control Board (CCB), use ESDIS own forms. For EMD CCBs, use the form available at the URL http://dmserver.gsfc.nasa.gov/forms/formindex.html. There are numbered items on the form that correspond exactly to the data entry required to be performed by the respective Configuration Management Administrator who maintains CCR records for the CCB on the distributed implementation of the Change Request Manager tool. Each CCB will have unique CCR identification sequence numbers. Each CCB can forward CCRs and reports from the Change Request Manager to SMC, which processes system-level CCRs for ESDIS CCB. The ESDIS CM Plan will determine the charter of the respective CCBs and thus the scope of CCR issues to be addressed by the site CCBs.

Earth Observing System Data and Information System (EOSDIS)
Core System (ECS)
Configuration Change Request (CCR)

1. **Configuration Control Board (CCB)**
ESDIS:___ ECS:___ SMC:___ DAAC: GSFC___, LaRC___, EDC___, NSIDC___,

2. CCR No.

3. Submitted Date:

4. Revision

5. Priority
☐ Emergency
☐ Urgent
☐ Routine

6. Change Class

7. Status

8. CCR Title:

9. Originator: Org: e-mail: phone:

10. Approval:______________________ signature ______________ date

11. Reason for Change

(indicate attachment ___)

12. Description of Change

(indicate attachment ___)

13. Impact Analysis:
Cost: ☐ None ☐ Small (Not exceeding $100,000) ☐ Medium ($100,000 to $500,000) ☐ Large (Over $500,000)
Evaluation Engineer: Org: e-mail: phone:
Impact Evaluators: ESDIS___; ECS Dev___; SEO___; SMC___; DAACs: GSFC___, LaRC___, EDC___, NSIDC___, EOC___;
Others____ (indicate attachment ___)

14. Comments: (Indicate Sites/ Organizations Affected)

(indicate attachment ___)

15. Board Action: ☐ Approved ☐ Withdrawn ☐ Disapproved ☐ Deferred Until_________ date
Further Action Required: ☐ ECP ☐ Waiver ☐ Deviation ☐ Tech Direction ☐ Contract Mod.
☐ DCN Other:____

16. CCB Approval
Chair:_______________ signature _________ date

17. CCR Implemented
CM Admin. signature:_____________ date:_____

Figure 9.2.4-1. Configuration Change Request (CCR) Form

The following enumerated text corresponds to the numbered items on the CCR form:

(1) **Change Control Board (CCB) --** The designated CCB is checked-off for changes processed by the ESDIS Change Control Board (CCB) and its EMD site-level chartered CCBs at the SMC, EOC, and DAACs (GSFC, LaRC, EDC, and NSIDC).

(2) **CCR Number --** The unique serialized CCR number is applied at each site.

(3) **Submitted Date --** The date that the CCR was prepared is documented.

(4) **Revision --** The current revision is designated for tracking changed versions of original CCR.

(5) **Priority --** The priority level of the CCR is assigned. Emergency CCRs may have already been implemented on a temporary basis by the Trouble Ticket Review Board (TTRB) with concurrence from the CCB Chair who later receives the CCR to document /implement the permanent change. Urgent items will be reviewed by the next CCB meeting. Routine items will be reviewed as soon as the schedule permits.

(6) **Change Class --** Change Classes are either I or II. Class I will be handled by ESDIS-only because of cost, schedule, and/or mission impacts that may require requirements changes. Class II items do not affect mission requirements, but may have cost and/or schedule implications which affect maintenance, operations, procedures, documentation, site-tailored items, COTS implementation, site installations of core system changes, science SW changes, etc.

(7) **Status --** Table 9.2.4-1 is a summary of the CCR states maintained by the Distributed Defect Tracking System (DDTS) application SW database tool that implements the Change Request Manager. Note that the hard copy form will not be updated chronically, but will be kept in the master suspense file of the CM Administrator until closed-out with a stamp (item #7 & 15) and appropriate signatures (viz., items 16, & 17).

Table 9.2.4-1. State Table Composition in DDTS Format

State Code	Available States	State Assigned
S	Submit	Submitted
N	New	New
A	Assign-Eval	Assign-Eval
O	Assign-Implement	Assigned-Implement
R	Implement	Implemented
T	Assign-Verify	Assigned-Verify
V	Verify	Verified
C	Close	Closed
D	Duplicate	Duplicate
F	Forward	Forwarded
P	Defer	Deferred

Explanations of DDTS' State Table Content:

Uppercase character in the 1st column is the character stored in the change request record to indicate what state a change request is in. DDTS uses this character to go into the table and extract the descriptive name for display in reports.

Names in the second column are the state in the present tense. They are shown on the DDTS list of states that are available for selection during input. It's also used by some of DDTS' query and report code. It facilitates querying based on descriptive names opposed to a single letter.

Names in the third column are the state in the past tense. They are shown on the DDTS change request record. It's also used by some of DDTS query and report code. It facilitates querying based on descriptive names opposed to a single letter.

Definitions (a state is the stage that a proposed change has reached in its life cycle.):

New - the initial state for all newly entered change requests.

Assign-Eval- state entered when the change request is being assigned to an engineer for evaluation/analysis.

Assign-Implement- state entered when the change request is being assigned to an engineer for development.

Implement-state entered when the proposed change has been developed.

Assign-Verify-state entered when the developed change is being assigned to an engineer for verification testing.

Verify- state entered when a developed change has been tested and verified that it functions properly.

Close- state entered when all activity specified in the change request has been completed or that the approval authority has decided to close it prior to completion of all activity.

Duplicate-state entered when a change request is determined to be a duplicate of an existing change request. Duplicate change request identifies change request being duplicated.

Forward-state entered when a change request needs to be forwarded to another DDTS defined project (In DDTS terminology, a project is a grouping of change requests. For example, a change request from a site project can be forwarded to an EMD project.).

Defer- state entered when activity on a change proposal has to be postponed.

(8) **CCR Title --** The CCR title is supplied by the originator.

(9) **Originator --** The originator name, organization, e-mail address, and phone number is given.

(10) **Approval --** The CCR is approved by the designated management authority that is assigned by the CCB. This sponsorship requirement acts as a primary filter to eliminate

from consideration those CCRs that cannot be implemented or which have no EMD site management support.

(11) **Reason for Change --** The reason for the change is narrated on the form and/or the designated attachment.

(12) **Description of Change --** The proposed implementation of the change is narrated along with any known impacts, resources, and expenses to be incurred.

(13) **Impact Analysis --** Impact analysis is documented in the form of Figure 9.2.4-2. The impact analysis is collected by the CCB Chair appointed Evaluation Engineer in coordination with the CM Administrator who maintains the CM records and assembles the review package for the CCB. The Evaluation Engineer documents the list of Impact Evaluators and derives and/ or verifies cost, technical, and schedule impact of the proposed change based on all inputs received. The results of the coordinated CCR Impact Analysis inputs are presented in the CCR Impact Summary form shown in Figure 9.2.4-3 part of the CCR review package.

(14) **Comments** -- Comments are added to the CCR to summarize sites and/ or organizations affected by the CCR. Additional comments may address proposed CCB dispositions and recommendations to be indicated by resolutions in item #15.

(15) **Board Action --** CCB actions and follow-up actions that will be facilitated and tracked by the CM Administrator are indicated. Possible CCB dispositions are given as approved, withdrawn, disapproved, and deferred (pending follow-up activities by the indicated schedule date). Further actions are indicated as:

Engineering Change Proposal (ECP)-- changed scope of contract requirements

Waiver-- declaration that certain contract requirements no longer apply

Deviation-- change of contract terms or substitution of terms or deliverable requirements

Technical Direction-- order by Contracting Officer’s Technical Representative (COTR) to perform certain tasks within the scope of the contract

Contract Modification-- changes to the terms of a contract

Document Change Notice (DCN)-- notification of changes to published documents

Others-- Engineering Change Notice, Change Order, Escalate to higher CCB authority, etc.

(16) **CCB Approval --** CCB approval signature authority by CCB Chair or designate.

(17) **CCR Implemented --** This signature and close-out stamp (item #7) are executed by the CM Administrator witnessing the completion of the CCR implementation process which

is tracked in the Change Request Manager automated tool DDTS and updated in Baseline Manager for affected version control status changes.

CCR Impact Analysis

Responder Request Number: ____
Responder:________________________
Responder Point of Contact:
address:______________________
____
phone:________________________
e-mail:________________________
CCB Schedule Date:__________
CCR Number:_________________
CCR Log Date:________________
CCR Originator:______________
CCR Originator Point of Contact:
address:______________________
____
phone:________________________
e-mail:________________________

Evaluation Engineer:_________________
Evaluation Engineer Point of Contact:
address:______________________
____
phone:________________________
e-mail:________________________
Requested Return Date:________________

Rough Order of Magnitude (ROM) Impact Analysis

Basis of Estimate:

Technical Assumptions and Comments:

Cost Impact:
None [ ]
Small [ ] < $ 100, 000
Medium [ ] $100,000 < x < $500,000
Large [ ] > $500,000

Schedule Impact:

Technical Assessment: (Your impact analysis should consider the implementation approach; interfaces affected; HW or SW changes required; documentation changes required-- change from/to pages; suggested alternatives, if any; and impact to security features. If your system is not impacted, please provide that information to the CM Administrator.)

Comments:

Signed:________________________
(Responder)

Date:_____________

Figure 9.2.4-2. EMD CCR Impact Analysis

CCR Impact Summary

Evaluation Engineer:__________________
Evaluation Engineer Point of Contact:
address:_______________________
____
phone:_________________________
e-mail:_________________________
CCR Board Date:_________________

Resources Summarized:

Technical Summary:

ROM Summary (BOE, Cost, and Schedule):

Recommendation:

Signed:_________________________
(Evaluator)

Date:______________

Figure 9.2.4-3. EMD CCR Impact Summary

9.2.4.2 Change Control Board Process (System and Site-level CCBs)

The EMD organization will provide administrative and technical support services for the CCB at each site. Each site's CCB is controlled by the host site organization and provides the authority and direction for the EMD contractor to modify the operational baseline. The ESDIS CCB has chartered an EMD Review Board to coordinate EMD system-level changes and problem management via the Sustaining Engineering Organization (SEO) contractor and on-site Review Boards that also act as site CCBs. This is illustrated using the CM Administrator's workflow for the SEO support of the EMD Review Board in Figure 9.2.4-4 and the On-Site CM Administrator's workflow for SEO support of the on-site CCB in Figure 9.2.4-5. The problem management process was discussed in detail in Chapter 8 of this document. Both diagrams illustrate the flow of CCRs through the respective CCBs with inputs from the review boards and evaluators that determine the disposition of proposed changes. Details of this process are given below:

System-level Change Control Procedures

(The enumeration corresponds to the diagram of Figure 9.2.4-4)

(1) Configuration Change Requests are received by the SEO CM Administrator from all sources with regard to the operational EMD Core System as described in Section 9.2.4.1. These changes designated as from other sources could involve system enhancements, procedures, interfaces (both external and internal), documentation changes, etc. that are not the subject of contemporaneous problem reports which would be first deliberated by the Trouble Ticket Review Board (TTRB) and/or Failure Review Board as explained below.

(2) Proposed common baseline changes will be proposed based on Trouble Ticket (TT) resolutions obtained from the respective review boards (see Chapter 8 for details). The respective TT would be closed via a corresponding CCR to either ratify, i.e., to make permanent the prior temporary/emergency action taken by the TTRB or to consider normal priority (scheduled) changes for incorporation into future change releases.

(3) The SEO CM Administrator is responsible for logging the CCR into the Change Request Manager.

(4) The CCB chair assigns an evaluator and the SEO CM Administrator coordinates impact assessment.

(5) Class I change requests (proposed changes that affect controlled milestones, schedules, budget, cost and requirements) are forwarded to the ESDIS CCB for consideration with recommendations from the EMD Review Board.

(6) Class II change requests (proposed changes affecting documentation, hardware [alternative use of], software [correction of errors], and COTS substitution without a Class I impact) are considered by EMD Review Board deliberations.

(7) Notice of proposed changes is distributed to affected parties and review board members to obtain and coordinate impact assessment and optimize the approach to implement proposed changes.

(8) The results of EMD Review Board deliberations are factored into review board resolutions which determine whether, when, or where the system changes will be implemented.

(9) Approved changes are processed by the SEO CM Administrator to the support activities, i.e., site CCBs, support personnel (SEO), vendors, etc. who are provided with change orders, schedule, and implementation instructions.

(10) Disapproved changes are processed by the SEO CM Administrator via official notifications, memo to the file, and update of the Change Request Manager (CRM).

(11) The SEO CM Administrator tracks implementation and closure of CCRs via directions to implementing organizations and their acknowledgements using the CRM tracking and statusing features.

(12) New versions and/or maintenance updates are annotated in Baseline Manager at the SMC and at the affected sites by following the procedures for configuration identification, activation dates, deactivations dates, and issuing version description documents.

(13) Simultaneously, the SW Change Manager (ClearCase) is updated with directory trees, installation files, and software as required by SW maintenance.

(14) Status of this activity to implement changes and assigned responsibilities is tracked through closure in the CRM at SMC and at the sites.

(15) The databases are synchronized by manual checking between applications Baseline Manager vs. CRM vs. SW Change Manager) and automated verification by the SW CM Manager for purposes of SW distribution and maintenance.

(16) The TT Review Board is empowered to make emergency fixes without common baseline changes and update these changes directly to Baseline Manager with documentation to follow via the CCR submitted to the appropriate CCB. Proposed common baseline changes must be submitted by CCR to the EMD Review Board.

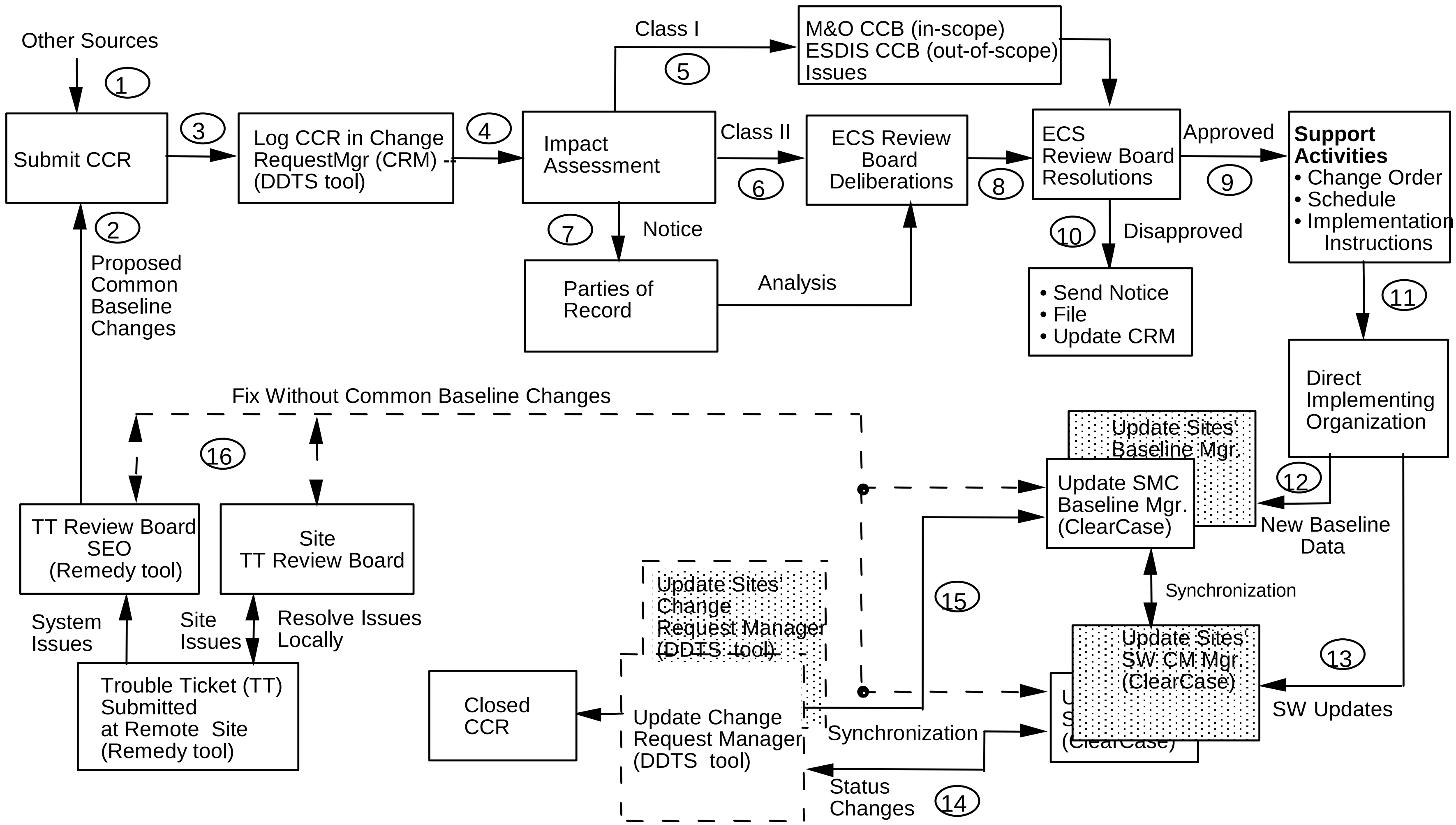

Figure 9.2.4-4. Work-Flow Diagram for SEO CM Administrator

Site-level Change Control Procedures

(The enumeration corresponds to the diagram of Figure 9.2.4-5)

(1) Configuration Change Requests are received by the Site CM Administrator from all sources with regard to the **site unique extensions** to the operational EMD Core System as described in Section 9.2.4.1. These changes designated as from other sources could involve system enhancements, procedures, interfaces (both external and internal), documentation changes, etc. that are not the subject of contemporaneous problem reports which would be first deliberated by the Site/ SEO Trouble Ticket Review Board (TTRB) and/or Failure Review Board as explained below.

(2) Proposed site baseline changes will be proposed based on Trouble Ticket (TT) resolutions obtained from the respective review boards (see Chapter 8 for details). The respective TT would be closed via a corresponding CCR to either ratify, i.e., to make permanent the prior temporary/emergency action taken by the TTRB or to consider normal priority (scheduled) changes for incorporation into future change releases.

(3) The Site CM Administrator is responsible for logging the CCR into the Change Request Manager.

(4) The CCB chair assigns an evaluator and the Site CM Administrator coordinates impact assessment.

(5) Class I/System Issues change requests (proposed changes that affect controlled milestones, schedules, budget, cost and requirements) are forwarded to the EMD Review Board for consideration with recommendations from the Site CCB. Class I issues are further forwarded with recommendations by the EMD Review Board to the SCDV CCB for in-scope issues and to the ESDIS CCB for consideration of out-of scope issues with respect to the SOW of the EMD Contract.

(6) Class II change requests (proposed changes affecting documentation, hardware [alternative use of], software [correction of errors], and COTS substitution without a Class I impact) are considered by Site CCB deliberations.

(7) Notice of proposed changes is distributed to affected parties and review board members to obtain and coordinate impact assessment and optimize the approach to implement proposed changes.

(8) The results of Site CCB deliberations are factored into CCB resolutions which determine whether, when, or where the system changes will be implemented.

(9) Approved changes are processed by the Site CM Administrator to the support activities, i.e., other CCBs, support personnel vendors, etc. who are provided with change orders, schedule, and implementation instructions.

(10) Disapproved changes are processed by the Site CM Administrator via official notifications, memo to the file, and update of the Change Request Manager (CRM).

(11) The Site CM Administrator tracks implementation and closure of CCRs via directions to implementing organizations and their acknowledgements using the CRM tracking and statusing features.

(12) New versions and/ or maintenance updates are annotated in Baseline Manager at the affected sites and the SMC by following the procedures for configuration identification, activation dates, deactivations dates, and issuing version description documents.

(13) Simultaneously, the SW Change Manager (ClearCase) is updated with directory trees, installation files, and software as required by SW maintenance.

(14) Status of this activity to implement changes and assigned responsibilities is tracked through closure in the CRM at the sites.

(15) The databases are synchronized by manual checking between applications Baseline Manager vs. CRM vs. SW Change Manager) and automated verification by the SW CM Manager for purposes of SW distribution and maintenance.

(16) The on-site TT Review Board is empowered to make emergency fixes without common baseline changes and update these changes directly to Baseline Manager with documentation to follow via the CCR submitted to the appropriate CCB. Proposed common baseline changes must be submitted by CCR to the EMD Review Board.

Each site's CCB accepts initial release or updates from the ESDIS CCB. Similarly, the Distributed Active Archive Center (DAAC) CCBs will accept product generation software from an ESDIS authority. Local tailoring and installation decisions are determined by the site CCB.

In the case of Evaluation Package (EP) and Prototype deliveries, the EMD CCB as directed by ESDIS will provide a configured, documented, executable with supporting files. Again, installation decisions are determined by the site CCB.

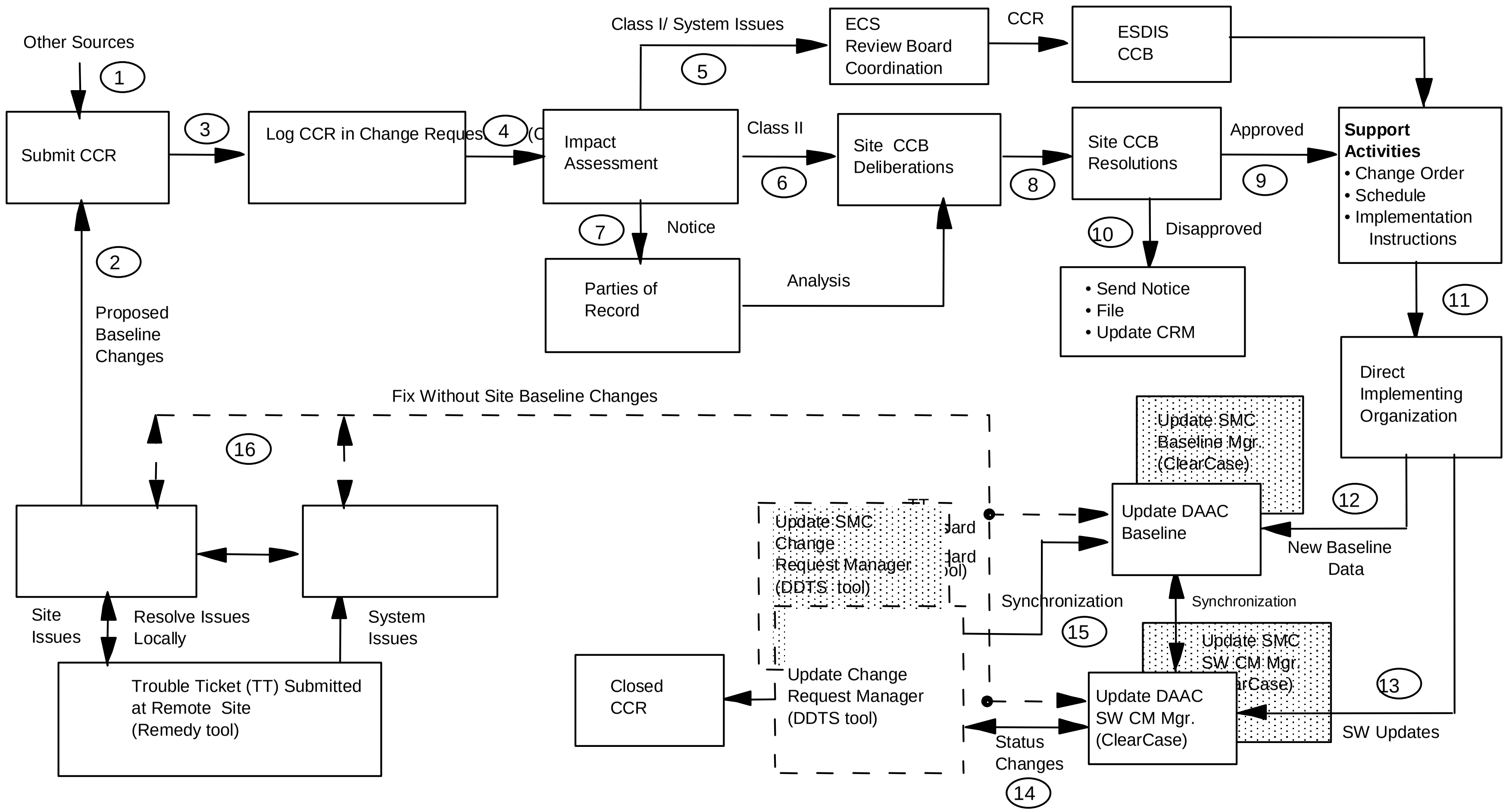

Figure 9.2.4-5. Work-Flow Diagram for Site-level CM Administrator

Each Science Computing Facility (SCF) is assumed to have a configuration control function. For commonality with other sites, it is assumed that this function will be performed by a CCB. A major difference, though, is the EMD contractor does not have an active role in the support of this CCB.

The SCF CCB will provide two types of configuration control:

(1) Configuration control of software and databases that are to be executed in another site's environment. In this mode, it operates very much as the EMD CCB does to establish a product baseline.

(2) Configuration control of SCF resources that are made available to the EOSDIS community. In this mode, its functions are the same as a DAAC CCB.

The EMD CM function at each DAAC will accept science SW and data items from the SCF CCB. These items will be incorporated into the DAAC's operational baseline as directed by the DAAC CCB.

The EOC CCB will control the operational configuration of the required EOC operational baseline. EMD CM will provide services as directed by the CCB.

The EMD Review Board will be charged with the responsibility for centralized coordination and control of EMD CM activities to ensure:

- EMD integrity and quality of service
- Successful coordination with both internal and external networks, systems, and on-site facilities
- Timely ESDIS CCB visibility into and oversight of EMD operations
- Convenient user administrative services

9.2.4.3 Configuration Control - Deviation and Waivers

1. Prior to completion of software development or to purchase of equipment or software, a Deviation from the specification may be granted by the ESDIS CCB.
2. Subsequent to the completion of development or delivery of equipment or software, a Waiver of specific requirements may be granted in order to accept defined nonconforming items. The waiver is traceable to a nonconformance report. A waiver is limited: Additional deliveries of like items must conform to approved requirements.
3. Departures from expected configurations that are not at variance with customer-approved requirements are handled as Nonconformance Reports.
4. A request for a Deviation or Waiver consists of a Deviation/Waiver Form shown in Figure 9.2.4-6 attached to a CCR form.

EMD Deviation / Waiver

Deviation/Waiver No.	Deviation ☐ Waiver ☐	Date:
NCR No.		
Title:		
Reason for Deviation / Waiver:		
Existing Requirement: (attach pages as needed)		
Departure: (attach pages as needed)		
Implementation Scope: (Identify CI, model, supplier, subcontractor, series, order no., location, release, time period, etc. as applicable)		

Document No.	Page/Paragraph Reference	Document Title

CM05MR95
EMD

Figure 9.2.4-6. Deviation/Waiver Form

5. Instructions for completing the form are listed below. Additional pages should be attached as necessary.

   Dev or Wai: Check the applicable box in accordance with the definitions given on page 1 of this Instruction.

   Waiver Number: Assigned by EMD CM Administrator

   Title: Enter a brief descriptive title. The title should be a statement, e.g., "Accept x in lieu of y".

   Reason: Describe the reason for the deviation or waiver. This may also include the history of the problem and consequences of not implementing the deviation/waiver.

   Existing Requirement: Specify and describe the baseline from which the deviation or waiver departs.

   Departure: Give instructions for the deviation or waiver with reference to the requirement.

   Implementation Scope: Include as appropriate, configuration item number, model no., supplier, subcontractor, series, serial numbers, order no., location, release number, quantity, time period or other criteria delimiting the deviation or waiver.

   Documents Affected: List current release number(s) of affected document(s).

6. The Deviation or Waiver CCR is prepared in accordance with the instructions in Section 9.2.4.1 with the following exceptions:

   Change Class: All deviations and waivers are change class I because they depart from approved requirements and must be approved by the ESDIS CCB.

   Description (Title): The title of the CCR will be the deviation or waiver title.

   Proposed Solution: Enter "See Attached Deviation/Waiver form."

7. Waiver and Deviation CCRs are submitted to the EMD CCB. Upon EMD CCB approval, the CM Administrator forwards the CCR to ESDIS CCB for authorization to implement the deviation or waiver.

8. When the deviation or waiver is authorized, the EMD CM Administrator immediately distributes the authorization information to the appropriate implementers and issues a document change order (DCO) to the Document Management Organization.

9. The CM Librarian copies the implementation instructions into the List of Deviations and Waivers at the front of the document and inserts the Deviation/Waiver number and effectivity at the point of applicability within the document.

10. Approved deviations/waivers are published via Document Change Notices (DCNs). Adding the deviation/waiver information to the document makes its status general knowledge. However, the deviation or waiver is in effect as soon as it is authorized by the customer.

11. A change in scope, effectivity or closeout, or any other change in a deviation or waiver requires a new CCR. The closeout or change is applied to the document via a Document Change Order in the same procedure as given in paragraphs 7 through 10.

9.3 Configuration Status Accounting Procedures

9.3.1 Purpose

Operational phase configuration status accounting (CSA) consists of recording and reporting information about the configuration status of the EMD Project's documentation, hardware and software throughout the Project life cycle. Periodic and ad hoc reports keep ESDIS informed of configuration status as the operational mission evolves. Reports to support reviews and audits will be extracted as needed starting from the RRR.

The Baseline Manager tool described in Section 9.7 records and tracks as-built products designated as EMD control items (i.e., custom, COTS, science, toolkit, (etc.) SW and HW items along with their associated documentation and records) and historical versions of EMD operational configurations. Baseline Manager, which is updated with the acceptance tested version of the EMD baseline at RRR, records and reports EMD document change status and histories, mission milestone baselines, and change status.

CSA entails maintaining version histories of delivered and maintained products as well as histories of operational baselines and changes made to each baseline. Additionally, CSA tracks the status of proposed changes from initial CCR submission to ultimate disposition and/or implementation. CSA also maintains historical records of CCRs.

9.3.2 Applicability

All ESDIS chartered CCBs

9.3.3 References

ESDIS CM Plan

EMD Configuration Management Plan 110-EMD-001

9.3.4 Procedures

The following are topical items subject to periodic or ad hoc reporting on behalf of the respective CCB or a system-level summary of information that will be reported by the CM Administrator representing the operational baseline for all the EMD sites.

(a) **New CCRs and Revisions.** This is a standard Change Request Manager report. This report will be issued monthly and summarized annually.

(b) **CCB Review.** Distribute CCR copies for review (and Impact Analysis forms if applicable). Print the agenda and distribute prior to the meeting.

(c) **Open Action Items.** Open action items should be statused regularly between meetings.

(d) **CCB Meeting.** Record the CCB's disposition of each CCR.

(e) **Record Action Items.** Record actions, assignments, and due dates.

(f) **CM Librarian Maintained Document Changes.** When all authorized document changes have been accomplished prepare DCN, post the final version on the EMD Document Data Server and distribute hardcopy as required.

(g) **Minutes Distribution.** Distribute minutes to the standard distribution and inform actionees of assigned action items.

(h) **CCR Implementation Status.**

- After CCB disposition, regularly status open CCRs until closure.
- Class I events include: CCR to EMD Review Board for review/approval; Technical Review Board; and ESDIS Disposition
- Further events are as follows for M&O implementation status: Consent Obtained; Item Received; Installed; Document Completed; etc.
- CCR CLOSED: A Class I CCR is not closed until the ESDIS contract officer's authorization is received or the reference CCR has been withdrawn.
- Class II document change CCRs may be closed with the CM Administrator's issuance of the DCN.
- Other non-document change CCRs may be closed when the originator verifies to the CM Administrator that all specified changes have been implemented.

9.4 Configuration Audits

9.4.1 Purpose

SEO will support Functional Configuration Audit /Physical Configuration Audit (FCA/PCA) by IATO at RRR. SEO will also support audits by ESDIS and our own Quality Office functions. Internal CM self-audits will be conducted by the SEO. Self-audits evaluate the Project's compliance with the Configuration Management Plan for the EMD Project and the ESDIS CMP. The CM self-audits will verify:

- That CM policies, procedures, and practices are being followed.
- That approved changes to documentation, and to software and hardware products are properly implemented.
- That the baseline documentation of each CI agrees with the as-deployed configuration and that adequate records of differences are available at all times.

A post-audit report is written outlining the specific items audited, audit findings, and corrective actions to be taken. All action items are tracked to closure.

In addition, SEO supports formal audits scheduled and conducted by ESDIS. These audits are conducted to validate that each EMD CI is in conformance with its functional and performance requirements defined in the technical documentation. The audits validate that:

- The as-built configuration compares directly with the documented configuration identification represented by the detailed CI specifications.
- Test results verify that each EMD product meets its specified performance requirements to the extent determinable by testing.
- The as-built configuration being shipped compares with the final tested configuration. Any differences between the audited configuration and the final tested configuration are documented.
- When not verified by test, the compatibility of EMD products with interfacing products or equipment is established by comparison of documentation with the interface specifications that apply.
- COTS products are included in audits as integral parts of the EMD baseline.

9.4.2 Applicability

All ESDIS chartered CCBs

9.4.3 References

Configuration Management Plan for the EMD Project	110-EMD-001

9.4.4 Procedures

The audits are standardized for a limited set of issues that drive the process for which the audit is taken, viz., FCA/PCA, Security Issues, General Accounting, Test Readiness Review, or Operational Certifications. The baseline references for the audits will be maintained by the ClearCase Baseline Manager tool (cf. Section 9.7). Release Notes documents will be used to document auditable changes to configured articles that are issued at the EMD configuration item (CI) level. Audit processes, including Project Instruction documentation, are discussed in the *Configuration Management Plan for the EMD Project*, 110-EMD-001.

The release notes will contain the prioritized current status summary of any Trouble Tickets/Discrepancy Reports against the CI that is being issued per the change request.

Some general guidelines and/or items that must be tailored for the specific size and scope of configuration audit to be conducted include:

(a) Audit Plan;

(b) Conference Agenda;

(c) Location to collect and analyze data; conduct meetings;

(d) Applicable specifications, drawings, manuals, schedules, design and test data;

(e) Test Results Analysis;

(f) Meeting minutes including resulting audit action items;

(g) Tools and inspection equipment necessary for evaluation and verification;

(h) Unencumbered access to the areas and facilities of incoming inspection, fabrication, production, and testing;

(i) Personnel from each engineering, production, and quality department to be available for discussion of their respective areas;

(j) Copies of inspection reports, process sheets, data sheets, and other documentation deemed necessary by the Government FCA/ PCA teams; and

(k) Isolation of the item(s) and detailed parts to be reviewed.

9.5 Archiving Procedures for the SW CM Manager (ClearCase)

9.5.1 Purpose

These instructions establish Configuration Management procedures for the backup of ClearCase Version Object Base (VOB) data, Views, and data delivered to the customer.

9.5.2 Applicability

All EMD CM Administrators, SW Maintenance Engineers, and Sustaining Engineers

9.5.3 References

EMD Configuration Management Plan	110-EMD-001
Archiving Procedures for ClearCase	CM-1-014-1
Delivery Package Archiving	CM-1-034-1

9.5.4 Definitions

<u>Build</u> - An assemblage of threads to produce a gradual buildup of system capabilities.

<u>EMD Development Facility (EDF)</u> - the software development environment including data, hardware, software, networks, facilities, and procedures used to support EMD software development and testing.

<u>Software</u> - for the purpose of this instruction, software includes all EMD-developed application software, COTS software, build and environmental instructions, and databases used in the execution of these products.

<u>System-level</u> - for the purpose of this instruction, system-level includes all EMD integration and test activities beginning with installation of software.

Software Development File - a repository for a collection of material pertinent to the development or support of software.

Thread - a set of components (software, hardware, and data) and operational procedures that implement a scenario, portion of a scenario, or multiple scenarios.

View - A unique workspace management or “sandbox management” that provides developers with transparent, file-level access to any version of any element through the use of dynamically-evaluated, user-specified version selection rules.

VOB - Versioned Object Base. Secure, permanent, mountable file system. Repository for storage of version-controlled data.

9.5.5 General

1. System Administrators maintain a backup for all EMD systems. This procedure documents an additional backup procedure for only ClearCase-specific data. This process is in parallel with System Administrator procedures. ClearCase, the EMD level software-tracking tool, contains within itself a number of repositories called Versioned Object Bases (VOBs). These VOBs maintain all versions of all software elements developed for the EMD project. Additionally, there are disk areas known as Views, which contain file(s) that may be newly created or modified but not yet given or returned to ClearCase control.
2. It is critical that these repositories, VOBs, get backed up synchronously on a daily basis, and that the backup tapes be maintained in such a way as to guarantee their restoration to the system in the event of a catastrophic failure. This CM archive procedure uses synchronization techniques to ensure that no data is lost. The System Administrator’s procedure does not include these techniques.
3. All EMD software is delivered electronically. Copies of the exact optical media that were delivered are made and stored similarly to in-house generated backup tapes.
5. Backups of VOBs are cumulative rather than incremental. Therefore, each tape has all data contained on previous archive tapes plus new files created, modified, etc.
6. A tape set consists of 31 archive tapes. The tapes are numbered from 1 to 31, a tape for each day of the month.
7. All software CM personnel will be familiar with the archive procedures.
8. All tape backup and archives are performed on the Unix host blood.

9.5.6 Procedures

1. A Software Configuration Management person will be assigned the responsibility for tape backup and archives. Daily backups are created every day at 02:30 a.m. via a UNIX cron job. Every workday morning the assigned Software CM person will perform tape archive procedures.

2. The daily backup tapes are obtained from and maintained in a fireproof safe that is kept locked. Key CM personnel hold the combination to the safe. The most current tape (the tape corresponding with the day of the month) is removed from the Triton tape drive and is always placed in the slot farthest to the left in the location marked “Current Backup”. The tape corresponding to the day of the month for the next day is inserted into the tape drive.
3. On the first business day of each month the backup tape from the last business day of the previous month is given to System Administrator personnel for transportation to an off-site storage facility. Periodically the tapes stored off-site are returned to CM for reuse.
4. All tapes will be labeled with the type of data contained.

   Example of a routine monthly backup tape label:

   Monthly ClearCase Backup Tape

   VOBs

   For month:Feb 2004

   Tar formatted.

   Point of Contact:

   Example of a Deployed Software CD:

   Patch_6A.08_SDSRV.01

   CCR: 03-0754
5. During weekends and holiday periods the tape for the current workday prior to the weekend/holiday will be placed in the tape drive. This tape will be written to on all days prior to the next workday. In this case should the system crash over the weekend/holiday period, there will be a more up-to-date backup. For example, on Friday, Friday’s tape would be placed in the drive. This tape will be written to on Saturday, Sunday, and Monday morning due to the automated cron job. Should the system crash on Saturday after 03:30 a.m., there would be a backup from Saturday available rather than Friday. On Monday this tape would be removed and replaced with the tape for Tuesday.
6. This process does not archive personal view storage areas (see Section 9.5.5, Item 3). Therefore, files checked out of ClearCase and under modification are not backed up.

9.6 Software Delivery and Installation

9.6.1 Purpose

This section describes the delivery of Sustaining Engineer Organization-developed software from the Landover facility to a remote site (a DAAC), and subsequent installation of the software onto target hosts, in accordance with configuration management controlled processes. The process begins with an approved CCR. The CCR precisely defines the software to be released to the remote sites. Software is prepared and delivered using the *DeliveryTool*. The *DeliveryTool* is a custom ClearCase tool that was developed to ensure accurate and controlled release of software. Software installation is controlled by each sites' Configuration Change Board (CCB). Landover CSG prepares and releases the software to the DAACs, but the DAACs control the installation of the software into their operational modes.

9.6.2 Applicability

All EMD sites' Sustaining Engineers, System Administrators, CM Administrators, and Maintenance Engineers

9.6.3 References

Configuration Management Plan for the EMD Project	110-EMD-001
COTS and Custom Software Preparation and Delivery	CM-1-032-1

9.6.4 Procedures

9.6.4.1 Overview

This section describes the release of CM controlled software under the authority of the SCDV CCB's approved CCRs.

Assumptions:

- CM maintains records of software preparations and deliveries.
- Baseline records are maintained in the Baseline Manager (ClearCase tool)
- Electronic Delivery is performed via Secure Copy (scp) to target DAACs.
- Resource Planning, Mode Management, and other issues are not addressed in this scenario.

Summary of Procedures:

- Software is prepared delivered using a CM operated *DeliveryTool*. An approved CCR specifies which previously prepared software is to delivered and to which sites. Refer to CM-1-032-1, *COTS and Custom Software Preparation and Delivery*
- DAAC CCB Approves the Installation of the delivered software into a DAAC Operational Baseline (e.g., a mode, "OPS", "TS1", etc.)
- Software is installed in accordance with CCR installation instructions, and in accordance with local DAAC procedures/processes
- Delivery Tracking pages are maintained by CSG, to include:

http://cmdm.east.hitc.com/baseline/

"Drop_6A.08_Tracking" – shows all custom code TEs, Patches, and Drops

"Engineering S/W" – shows all Engineering Software deliveries

"COTS S/W Tracking" – shows all COTS software electronic deliveries

9.6.4.2 Functional Roles

SEO CM Administrator--Ensures that changes to the hardware, software, and procedures are properly documented and coordinated. Recommends configuration of all EMD configured hardware and software.

DAAC/SMC CM Administrator--Ensures that changes to the hardware, software, and procedures are properly documented and coordinated. Maintains control of all configured hardware and software. Assists in the development and administration of the library with respect to local configuration management procedures.

DAAC Sustaining Engineering--SW Maintainer--Produces, delivers, and documents the corrections, modifications, and enhancements made to EMD software (including COTS), and/or adaptations or incorporations of EMD COTS software.

9.7 Baseline Manager

9.7.1 Overview

The EMD provides a *ClearCase* Baseline Manager (BLM) tool to assist in documenting changes to the baseline, and to maintain a historical record of those changes. The BLM tool is used at the EMD Development Facility to maintain system-level records and site-level records; baseline reports are accessible at the operational sites by accessing the URL http:/cmdm.east.hitc.com/baseline/. This is the EMD Baseline Information System. Selection of the "Technical Documents" button provides detailed reports regarding the EMD baseline.

Baseline Manager (BLM) is used to record and report the CCR approved operational baseline of EMD products. It contains the configuration records for each product baseline. It identifies products by CI name, description, location, version, and component configured articles. It also provides traceability to previous configurations. The BLM tool is an extension of the *ClearCase* COTS software. It provides access to functions for maintaining control item and bill of material information. The baseline data may only be affected by approved CCRs.

9.7.2 Baseline Terms and Concepts

Baseline management is a process to identify and control versions of hardware and software, to provide a standard configuration of systems throughout all sites, and allow unique site-configured systems and baselines. It identifies interdependencies between hardware and software items, and permits maintenance of a complete history of baseline changes throughout the life of the project. For EMD baseline management and BLM tools, certain terms and concepts are key to understanding how data on the system baseline are stored and tracked.

Control Item – any EMD item under version control by Configuration Management.

Configuration Item – an aggregation of hardware, firmware, software, or any discrete component or portion, which satisfies and end user function and is designated for configuration control.

Baseline – a configuration identification document or set of such documents formally designated by the Government at a specific time during the life cycle of a configuration item (CI).

Configured Article – a control item reportable as part of the Configured Articles List (CAL).

CIL – a Configuration Items List (CIL) identifies the approved set of CIs that are subject to CM requirements and procedures.

CAL – a Configured Articles List (CAL) describes all CIs, critical item hardware and software, and supporting documentation by which the exact configuration definition of the hardware and software can be determined.

Additional terms, some of which address specific entries in the BLM tool, further define how data on the system baseline items and structure are tracked.

Assembly – An item made up of other items.

Bill of Material – The list of items that comprise an assembly.

Product Structure – The parent-child pairings that define the bill of material for an assembly; each product structure record specifies the effective dates and quantities for a single component of a parent for each engineering change.

Active Date – The date a component becomes effective in an assembly's bill of material. This date is the CCR approval date.

Inactive Date – The date a component is no longer effective in an assembly's bill of material. This date is when a new CCR displaces or affects the item.

Engineering Change – A mechanism for grouping, reporting, and controlling product changes collectively.

Revision – The sequence number of a product structure change to an assembly; it signifies a change to the configuration of an assembly that does not alter its form, fit, or function.

Implementation Status – A record describing the deployment of a control item to a site and the current state and associated date of its implementation; each control item has one record for each site to which it is deployed.

Exporting Data – Creating a formatted file or records extracted from the BLM database; control item engineering change, product structure, and interdependency records may be extracted and sent to other application software at any site.

Importing Data – Loading BLM data from a formatted file.

At the lowest level, the baseline is composed of configured articles that are the specific types of items that make up EMD and are tracked using the BLM tool. It is important to recognize, however, that we impose a conceptual structure on those configured articles to help us think about the system. In fact, it is possible to conceptualize the structure of the system in a number of different ways, and we may select a different conceptual structure based on the requirements of the situation. The EMD baseline management approach and the BLM tool permit recording and tracking these different conceptual baselines, which can be related to the same records of the configured articles.

For example, system designers may conceptualize the system in terms that will help them track subsystems and the configuration items for which each subsystem team is responsible. This may produce a baseline structured according to a design view, such as that shown in Figure 9.7.2-1.

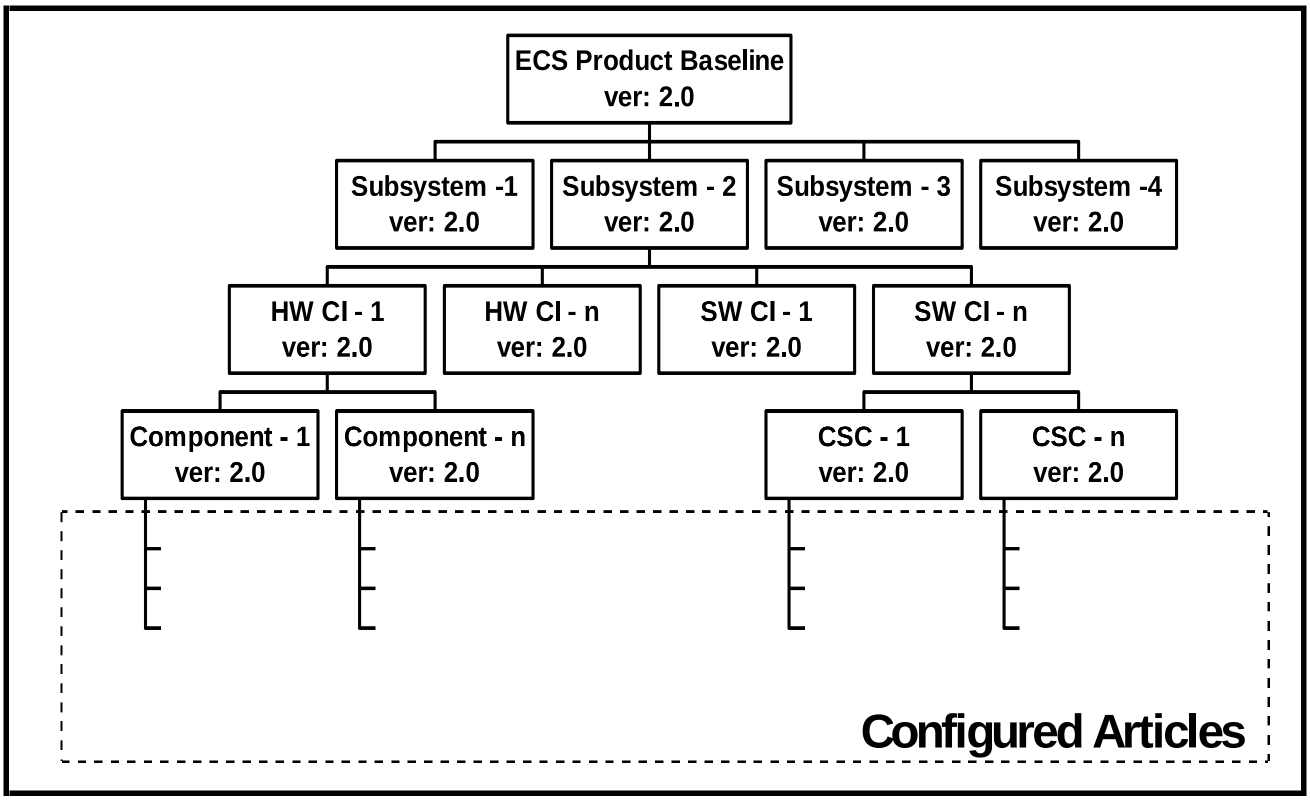

Figure 9.7.2-1. EMD Baseline Concept from a Design (CIL/CAL) View

At an operations site, the concept reflected in the upper layers of the Design View baseline structure may not be particularly useful. Although the same configured articles are involved, it may be desirable, for instance, to track items from the viewpoint of network administration. The resulting baseline product structure may reflect that shown in Figure 9.7.2-2.

Even if an operations site is to view EMD product structure as composed of subsystems, it is likely that the concept of CIs will be of little use. Instead, the site is likely to be focused on what hosts make up the subsystems. Therefore, the subsystem view at an operations site may be similar to that illustrated in Figure 9.7.2-3.

The Baseline Manager database implemented at the EMD Development Facility reflects EMD-developed product structures, and site personnel may not normally need all the data necessary to define these product structures. Instead, BLM tasks are likely to be limited to areas such as noting system changes, perhaps in the context of site-unique requirements and data. However, an understanding of the different ways of conceptualizing the system will help in interpreting baseline data reflected in the BLM.

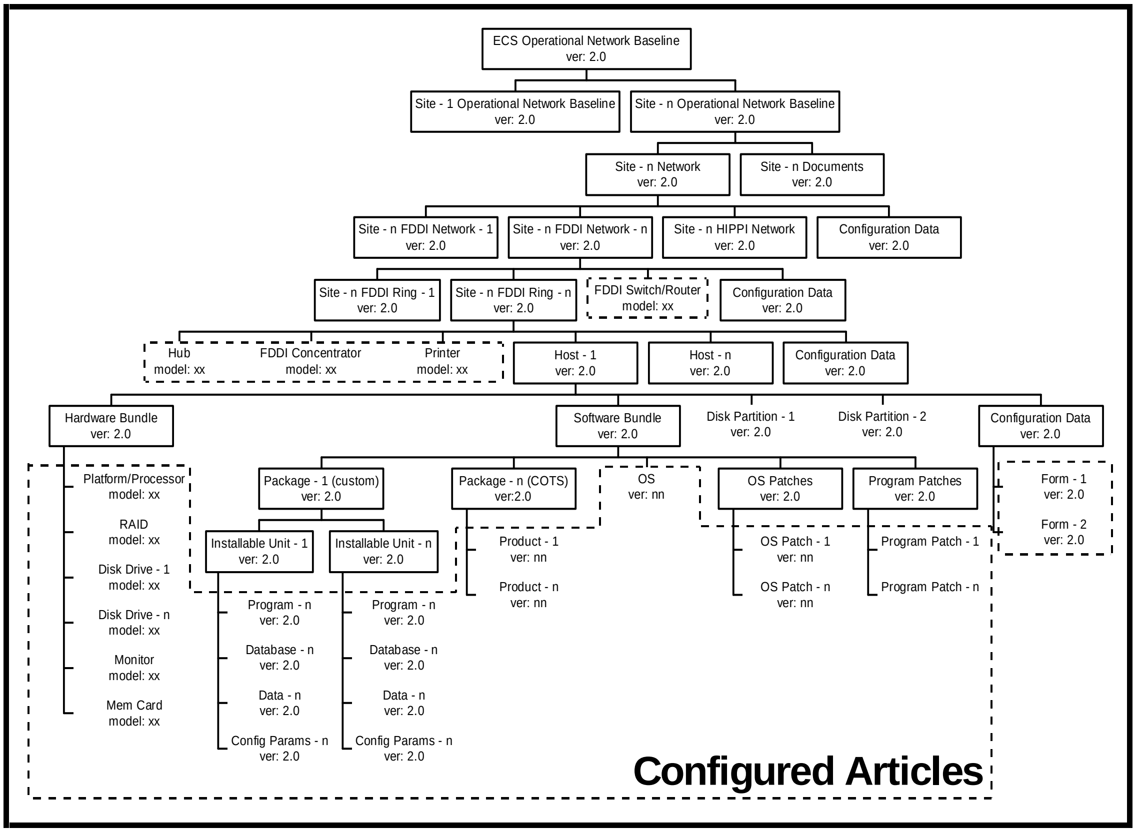

Figure 9.7.2-2. EMD Baseline Concept from an Operational (Network) View

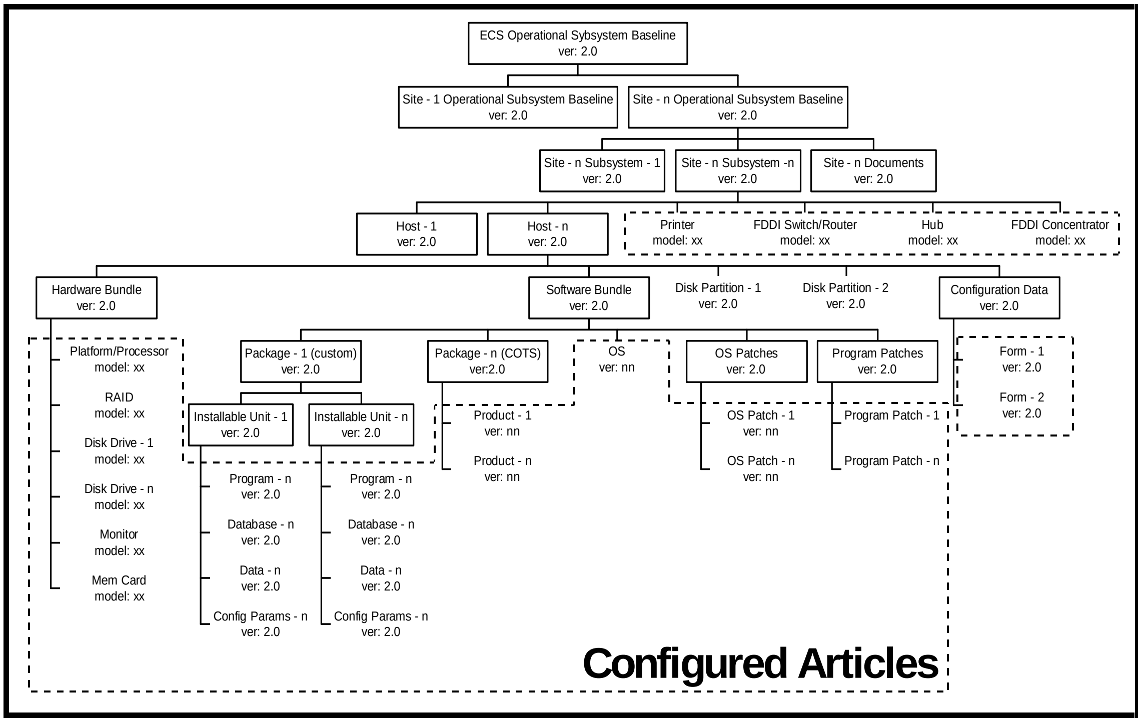

Figure 9.7.2-3. EMD Baseline Concept from an Operational (Subsystem) View

9.7.3 Baseline Manager (BLM) Outputs Useful at the Sites

The BLM manages the COTS software, operating system patches, and COTS software patch baselines. The BLM records, including information on all scripts, data, and GUIs, are maintained and managed at the EMD Development Facility using ClearCase. The BLM tool produces some of the 910/920 Technical Document reports, with automated posting to EMD Baseline Information System (EBIS) and replication to server **cmdm.east.hitc.com**. The reports include the following documents that affect all sites:

- COTS Software Versions 910-TDA-003.
- Site-Host Map 910-TDA-005.
- Critical COTS Software List 910-TDA-023.
- COTS S/W Where-Used Reports 910-TDA-030.

The reports also include the following documents that are site specific:

- Hardware-Software Maps 920-TD(*x*)-002.

  (*Note*: The *x* represents a letter designating specific DAACs. *e.g.*, g = GSFC, l = LaRC, e = LP DAAC, s = SMC, v = VATC, p = PVC, and n = NSIDC.)

- Hardware-Patch Maps 920-TD(*x*)-014.

All BLM records are related to approved Configuration Change Requests (CCR) and Release Notes documents (i.e., series 914-TDA-xxx).

The Configuration Management (CM) organization is the principal user of the BLM tool, using it to implement changes to the baseline. The system is used daily to describe CCB-approved system components and to track sites and machines where version-controlled items are configuration controlled. In addition BLM supports other functions such as configuration audits, system engineering and deployment activities. The BLM records describe the hosts for each site. The sites are the operational Distributed Active Archive Centers (DAACs), Verification and Acceptance Test Center (VATC) and Performance Verification Center (PVC). The system also tracks the COTS software and patches that are mapped to their respective hosts. EBIS accommodates the identification of all configuration-controlled items such as documents, and disk partitions.

The BLM capabilities are used to:

- maintain records that identify what items comprise individual, baselined, system configurations
- identify the versions and variants of hardware and software items that are currently baselined together with the assemblies (e.g., hosts, subsystems, and networks) that use them
- record item interdependencies and the sites to which baselined items are deployed
- keep chronological histories of baseline changes and traceability of items to predecessor versions and system releases

9.7.4 Procedure for Retrieving Baseline Reports

When the EMD software baseline is changed (e.g., addition of a script, update or replacement of a Graphical User Interface (GUI) package), the change must be reflected in the collection, or "catalog," of control items that make up the affected Computer Software Component (CSC) assembly in the EMD product structure. In the BLM software at the EDF, to document the change it is necessary to add the new element to the catalog of version-controlled items, define an engineering change for the CSC assembly, and include the element in the list of items that will now make up that assembly. Once the change is documented, baseline reports reflect the new information. These reports may be accessed through the EMD CMDM server for the EMD Baseline Information System (EBIS). The EBIS page is obtained by a click on the **EMD Baseline** button at the end of the Tools line near the top of the EMD CMDM Server page. On the EBIS page, the EMD Baseline Information Technical Documentation is accessible through use of the **Technical Documents** button at the top of the row of buttons on the left side.

The resulting **EMD Baseline Technical Documentation** page lists the document series, title, and document number. The document numbers are links that provide access to the listed documents. The titles of some documents indicate BLM origin by inclusion of the parenthetical notation **(ClearCase)**.

The following procedure is applicable for retrieving baseline reports.

1. At the UNIX command shell prompt, type **setenv DISPLAY *clientname*:0.0** and then press the **Return/Enter** key.
   - For ***clientname***, use either the local terminal/workstation IP address or its machine name.
2. Start the log-in to a Netscape host by typing **/tools/bin/ssh *hostname*** *(e.g.*, g0ins02, e0ins02, l0ins02, n0ins02) at the UNIX command shell prompt, and press the **Return/Enter** key.
   - If you receive the message, **Host key not found from the list of known hosts. Are you sure you want to continue connecting (yes/no)?** type **yes** ("**y**" alone does not work).
   - If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter passphrase for RSA key '<*user@localhost*>'** appears; continue with Step 3.
   - If you have not previously set up a secure shell passphrase; go to Step 4.
3. If a prompt to **Enter passphrase for RSA key '<*user@localhost*>'** appears, type your ***Passphrase*** and then press the **Return/Enter** key. Go to Step 5.
4. At the **<*user@remotehost*>'s password:** prompt, type your ***Password*** and then press the **Return/Enter** key.
   - You are logged in and a UNIX command shell prompt is displayed.
5. Type **netscape &** and then press the **Return/Enter** key.
   - The Netscape web browser is displayed.
6. Click in the **Netsite:** field.
   - The field is highlighted.
7. Type the Universal Resource Locator (URL) for the EMD Baseline Information Server (**http://cmdm.east.hitc.com**) and then press the **Return/Enter** key.
   - The **EMD CMDM Server** home page is displayed, offering access to EMD Baseline information as well as a number of tools, EMD web sites, and NASA EOS web sites.
8. On the Tools line click on the **EMD Baseline** button.
   - The **EMD Baseline Information System** page is displayed
9. Click on the **Technical Documents** button.
   - The **EMD Baseline Technical Documentation** page is displayed.
10. Locate the desired report, scrolling down as necessary.
    - Reports derived from the BLM may be indicated with a parenthetical notation **(ClearCase)** in the title entry.
11. Click on the link for the document to be accessed.
    - A directory is displayed with one or more document numbers and versions indicated as links.

12. Click on the link for the document and version desired.

    - The document is displayed, and can be printed and searched.

10. Metadata Administration

Every science data product generated and archived by the ECS must be described to the system by metadata which are put into an inventory and then used to retrieve and distribute the data to users of the system. The ECS Earth Science Data Model, described in documents 420-TP-020-001 and 420-TP-016, organizes the metadata into groups of related attributes and services to be performed on the data products. These "core" attributes are necessary to identify, interpret and perform services on granules and collections. The Data Model also provides for "product-specific" attributes, i.e. attributes which are unique to a specific data product.

The smallest aggregation of data that is independently described and inventoried in ECS is referred to as a data granule. Granules are organized into logical groupings called collections in which the granule metadata varies principally by time or location, called single-type collections.

Every collection is described by an Earth Science Data Type (ESDT) and is made known to the system by an ESDT descriptor file and associated software code which is built into the Data Server's dynamic link library (DLL) to perform the services.

Metadata administration includes creating and updating ESDTs within ECS. Collections may be modified and updated over time. In addition, quality assurance is performed after ESDTs have been installed and granules have been generated and stored in the archives. Collection-level metadata can be updated by updating the ESDT. Granule-level metadata can be updated manually (i.e. not as a result of an operation such as Subsetting which modifies the science data content of a granule) by setting the Quality Assurance flags and explanations. Procedures for updating these flags are provided in Section 15.

10.1 ESDT Descriptor Files

The primary task in establishing a collection is providing the core and product-specific metadata attribute values. This is done by creating an Earth Science Data Type (ESDT) descriptor file. The descriptor file is also used to specify the data services that are available for granules that belong to the collection. These services are implemented as methods of a Dynamic Link Library (DLL) containing C++ code to accomplish each service. The descriptor file and the DLL are the means by which a collection is made known to the Science Data Server (SDSRV).

The ESDT descriptor is composed of sections, all in ODL, containing the following information:

- Collection level metadata attributes with values contained in the descriptor.
- Granule level metadata attributes whose values are supplied primarily by the Product Generation Executives (PGEs) during runtime.
- Valid values and permitted ranges for all product-specific attributes.
- List of services for all the granules in the collection and events which trigger responses throughout the system.

The ESDT descriptor file is created by Metadata Works (MDWorks), discussed in Section 10.3.3.

The services that apply to a collection are specified in the ESDT descriptor file. Metadata Works automatically inserts standard ECS-supplied services such as insert and search into the descriptor file. Product-specific services, such as Subsetting or a product-specific acquire, require executable code to enact those services. This code is contained in the DLL. The DLL is written and tested by either the ECS developers or sustaining engineering personnel at the DAAC.

After the ESDT (both descriptor file and DLL) has been generated it must be installed on the Science Data Server before the first data granule can be inserted. During this installation process, information from the ESDT Descriptor File is propagated to the Data Management and Interoperability subsystems, and to the Subscription Server, which must all be operating during the ESDT installation process. The detailed procedures for ESDT installation into ECS are described in Section 11.3.

10.1.1 Steps in Generating a Descriptor File

ESDTs for Distributable Product

These are the typical steps used in generating a descriptor file:

Identify desired collection-level metadata attributes

- For permanent and interim files use only the minimum attributes.
- For distributable products identify all applicable attributes. This will involve reading appropriate documentation and interacting with the data provider.

Identify granule-level attributes

- If a sample metadata configuration file is available from the data provider, use this.

Check valids for core attributes (write CCR if new valids are required).

Check PSAs (register PSAs if new).

Gather metadata into a spreadsheet, or use Metadata Works to enter metadata directly.

Use custom built scripts to generate the descriptor file from the spreadsheet, if used.

Verify the descriptor file as outlined below.

Check descriptor files into ClearCase.

Notify the DLL Team Lead of the newly prepared descriptor files.

10.1.2 Verifying Descriptor Files

- Run the PERL script "update.pl", following the instuctions in the script prologue. (This script makes sure that the inventory metadata attributes are all listed as event qualifiers in the EVENT group.)
- Run the PERL script "esdtQC.pl", following the instructions in the script prologue. Make any necessary corrections in response to errors issued, and rerun. Repeat until there are no errors. (This script checks for more than 30 common descriptor file errors.)
- Run the PERL script "required.pl", following the instructions in the script prologue. Add any missing attributes as indicated.
- Run the "testodl.csh" utility to ensure that there are no errors in the ODL structure for the descriptor file. Make any necessary corrections in response to errors issued, and rerun. Repeat until there are no errors.

10.2 Preparation of Earth Science Data Types

An ESDT goes through pre-operational life cycle steps starting with an analysis of the collection's need and continuing through development and operational installation. This process involves actions by the Data Provider or User in addition to ECS. These procedures are detailed in Project Instruction SO-1-002, "Generation and Change of ESDTs, Valids, and PSAs." The overall workflow is shown in Figure 10.2-1.

10.2.1 Definitions

Archive - A File Type indicating granules will be inserted to Data Server for long-term storage and acquisition for distribution.

Full - A level of metadata coverage intended for data products which are produced within ECS.

Collection - A related group of data granules.

Granule - The smallest data element which is identified in the inventory tables.

Interim - A File Type indicating granules are temporarily stored in support of product generation.

Intermediate - A level of metadata coverage intended for contemporaneous data products which are not produced within ECS.

Limited - A level of metadata coverage intended for heritage data products brought into ECS for distribution

Minimal -A level of metadata coverage sufficient to uniquely identify a collection or granule.

Permanent - A File Type indicating static or semi-static granules which are used only as inputs in product generation.

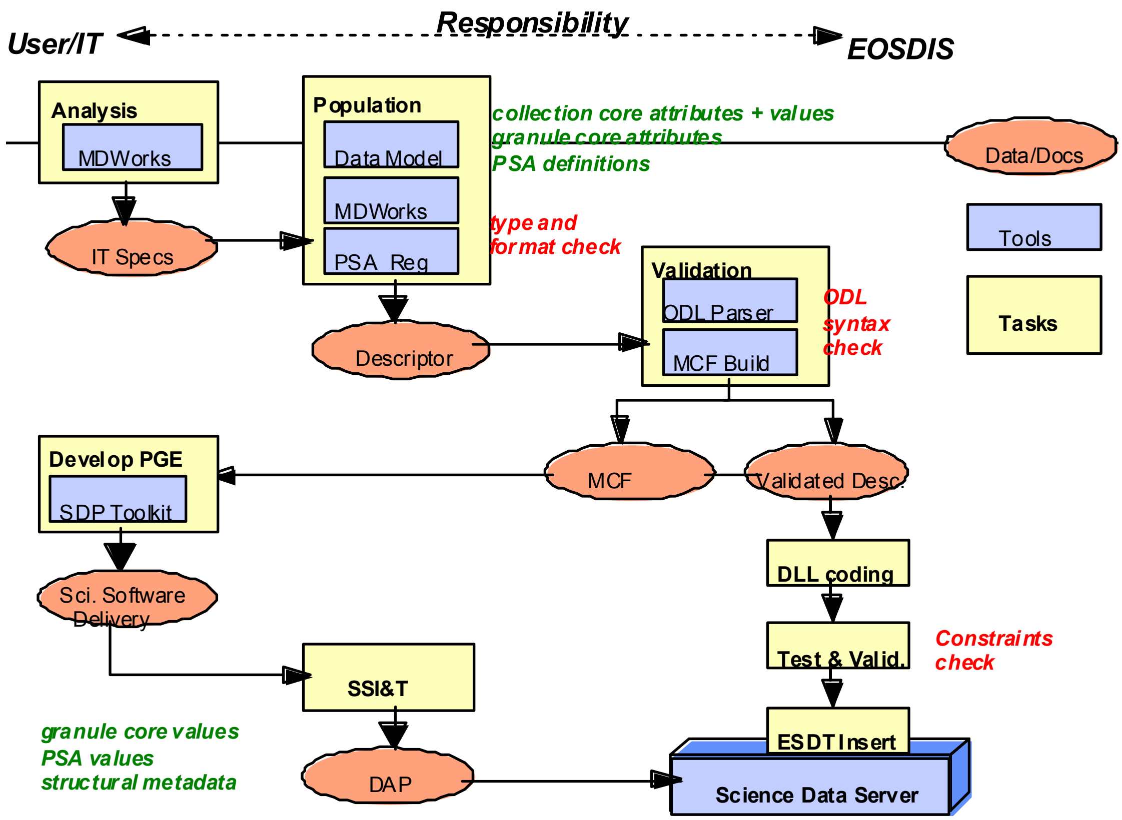

Figure 10.2-1. The ESDT Generation Process

Product Specific Attributes - Attributes that are defined by the data provider in support of searching for specific granules

Valid - An allowable metadata value.

10.2.2 Process

1. Need Analysis

- The baseline list of science ESDTs and their services is controlled by the ECS CCB. This baseline was established through an analysis of the ECS Functional and Performance Requirements Specification, the Technical Baseline established from inputs from the Ad Hoc Working Group on Production, and meetings with the individual data providers to define the basic requirements of each ESDT.

  These basic requirements are:

  - Data Provider File Designation,
  - File Type (Permanent, Interim, Archive)

- Level of Metadata Coverage (Minimal, Limited, Intermediate, Full)

- For new ESDTs not currently in the development baseline, the result of the Need Analysis forms the basis for approving the inclusion of the ESDT into ECS. This is accomplished through the CCR process.

2. ESDT Specification

- This step results in a set of specifications extending the results of the needed analysis and providing the information needed to implement an ESDT. This step is executed only if the ESDT has been included in the baseline. The roles and responsibilities for developing the specification are as above.

  The specifications must include:

  - ShortName and VersionID of the ESDT
  - A list of the metadata attributes needed, valids, and any constraints on attributes. This list is to be drawn from the B.x Descriptor File Template. This template, under CCB control, is based on the attributes defined in "Release-B SDPS Database Design and Database Schema Specifications" [311-CD-008-001] (i.e., DID 311), as modified for B.0 (for example) by "B.0 Earth Science Data Model " [420-TP-020-001].
  - A list and specification of the services needed (e.g., specification of the INSERT, SEARCH, ACQUIRE and SUBSCRIPTION semantics).

3. ESDT Generation

- Once the ESDT Specification has been developed and the applicable attributes identified, the necessary metadata has to be gathered, the metadata values checked against the valid values and the product-specific attributes (PSA) need to be checked against the list of PSAs that are already defined (see Fig. 10.2-2).

- Once the collection-level metadata and granule-level attributes have been checked, then the descriptor file is generated, the Dynamic Link Library produced, and testing and validation of the ESDT performed. This process is further elaborated in the sections below.

- For a one-of-a-kind, distributable product with Full metadata coverage, this process can take up to six weeks to accomplish. For a related group of products with identical services, much of the Descriptor File and DLL of the first ESDT can be reused, and the cycle time for preparing subsequent ESDTs in the related group is much less.

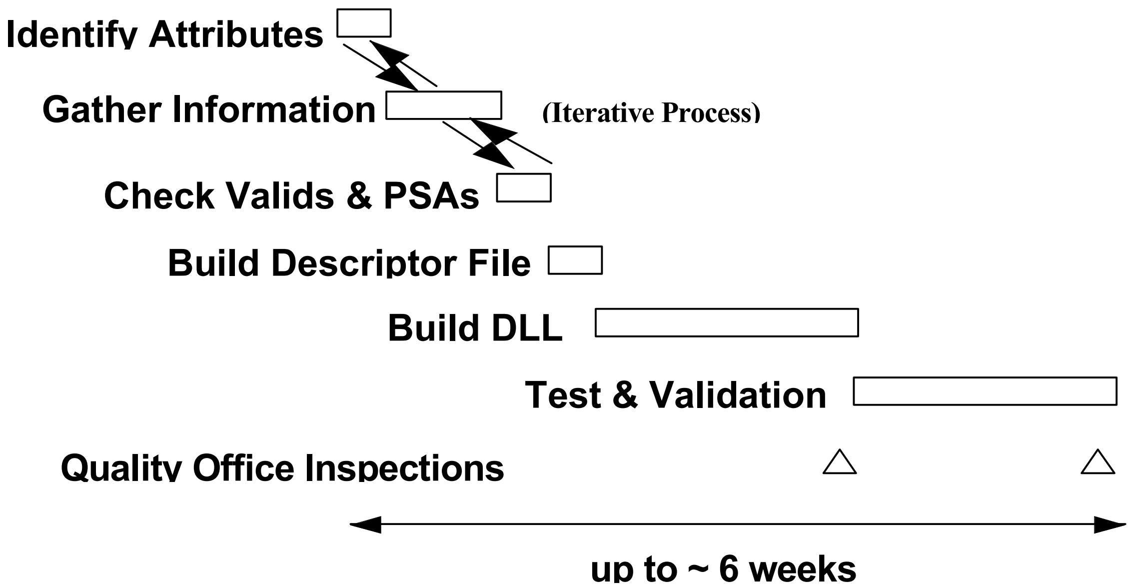

Figure 10.2-2. Steps in ESDT Development

10.3 Tools Used in Generating a Descriptor File

Figure 10.3-1 shows the some of the tools that have been developed, and indicates supporting information flows between these tools. Brief descriptions of these tools are provided below.

10.3.1 Data Dictionary and Valids Checking

Each proposed new valid must be reviewed by the Data Modeling group in order to validate that they fit within the model. When approved, within a couple days at most, the new valid will be available for use.

10.3.2 PSA Registry

Each Product Specific Attribute must be registered to ensure that the name is uniquely defined across all products and that no two names are applied to the same definition (aliases may be applied at the user's Client and in Data Management, but should not be applied within the inventory). An X-Windows based GUI has been developed to assist in the registration process.

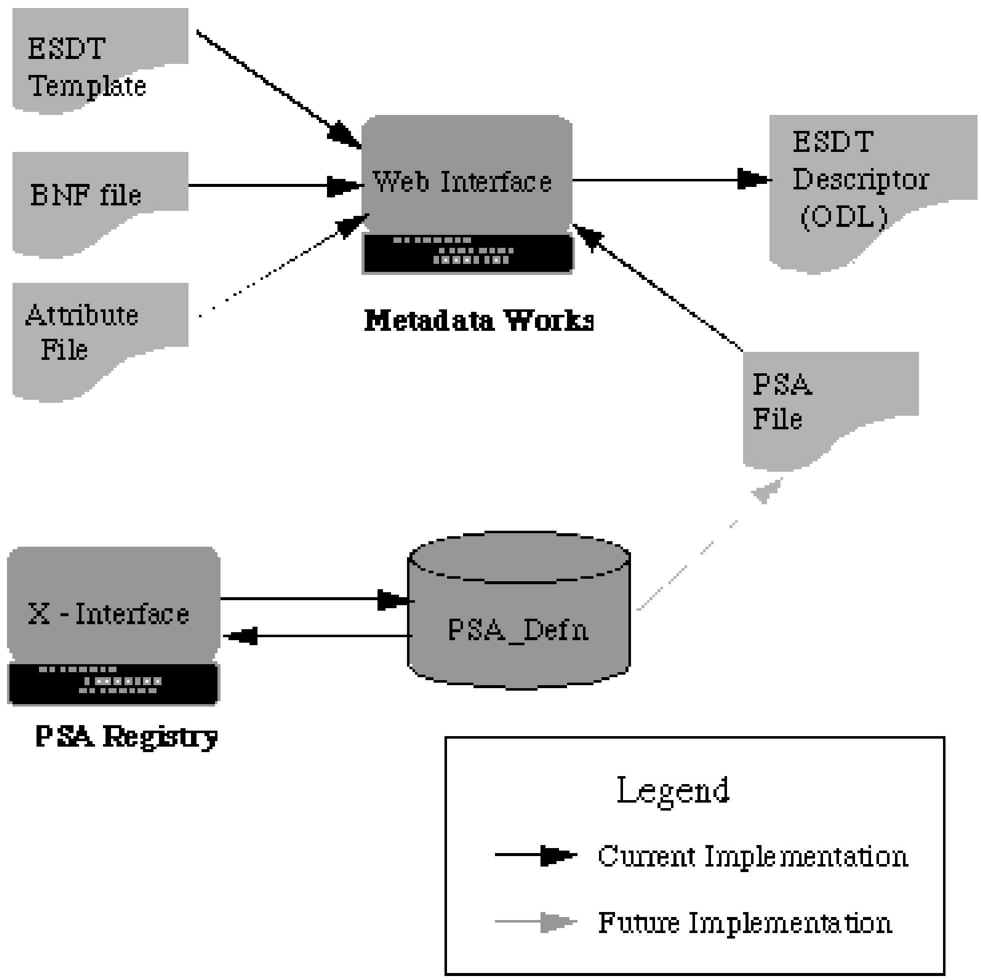

Figure 10.3-1. Tools and Supporting Information Flows

10.3.3 Metadata Works

To support the entry of metadata for ESDTs with Full or Intermediate coverage, an HTML-based GUI has been developed. Metadata Works allows the person entering the metadata to start with a completely empty form, or to define a new descriptor file employing defaults of already defined groups or entire previously defined descriptor files. When the metadata entry is complete for a descriptor file, Metadata Works is then used to generate a complete Descriptor file employing the proper ODL syntax.

10.4 Metadata Population

10.4.1 Collection-Level Metadata

A majority of the attributes in the ECS Data Model apply to all the granules in the collection. These are known as collection-level attributes. There can be both core and product-specific collection-level attributes, defined once prior to establishing the collection.

Collection-level metadata is input via an HTML-forms GUI tool called Metadata Works, described further in Section 10.3.3. Based on the ECS Data Model it is designed for use by the data provider; e.g., an instrument team scientist or other person having extensive knowledge of the data. It can import as defaults the attribute values from a collection that has already been populated. A sequence of screens is presented to the user enabling specification of all required and optional attributes, with a list of permitted values presented where appropriate. Help screens give attribute definitions, data types and other relevant information to assist in the specification.

10.4.2 Granule-Level Metadata

The attributes in the ECS Data Model which can vary on a granule-by-granule basis are known as granule-level attributes. There can be both core and product-specific granule-level attributes.

Granule-level metadata are specified and populated using the Metadata Configuration File (MCF). The MCF is derived from information contained in the ESDT descriptor file and delivered by the Science Data Server for use by the Science Data Processing or Ingest Subsystems. The MCF specifies how the searchable metadata attributes will be populated in the SDSRV database. For data products generated by ECS, the science software or Product Generation Executive (PGE) interacts with the MCF using metadata tools contained in the Science Data Processing Toolkit. Through this process values are set for metadata attributes specified in the "source" MCF, such as the temporal or spatial coverage of each granule. These values are then inserted into a "target" MCF at PGE run time. The MCF is used in a similar manner for data entering ECS through the Ingest subsystem.

Procedures for entering data into ECS through the Ingest subsystem are described in Chapter 16. "Ingest". Procedures for running a PGE are described in Chapters 11.10 "Running a PGE in a Simulated SCF Environment at the DAAC" and 11.13 "PGE Planning Processing and Product Retrieval".

The Inventory Metadata section of Metadata Works is used to capture granule-level metadata specifications. An unofficial MCF may be generated as output from Metadata Works, for testing purposes. Final testing should always be done with the MCF provided by ECS which is guaranteed to be identical to the one delivered by SDSRV at run time.

The actual population of the granule-level attribute values into ECS inventory data bases takes place during the insert of a data granule into the SDSRV. Each data granule consists of one or more physical files. Accompanying each granule is a metadata record; i.e., an ASCII file containing the granule level attributes and their values in ODL. Only one metadata record is allowed per granule, i.e. no. sub-granule records are allowed, and no metadata records are shared between granules.

Procedures used to initiate the running of PGEs are described in Chapter 11.13 "PGE Planning and Processing".

10.4.3 Product-Specific Metadata

Product-specific metadata can be at both the granule-level and the collection-level. Product-specific metadata may (at the data providers election) be contained in the inventory tables in the database, in which case it will be searchable by ECS. There is also a provision to store product-specific metadata within granules that is available only when the granule has been ordered and delivered. This is termed archive metadata and is specified in a separate ODL group in the MCF.

In the granule metadata, the core attribute that is available to store product-specific metadata is called ParameterValue. The metadata describing this attribute is specified by the data provider through the AdditionalAttributes class at the collection-level. The units of measure, range, accuracy, and resolution for this is specified in the PhysicalParameterDetails class, also at the collection-level.

Product-specific metadata at the collection level is specified with Metadata Works at the time the other collection level metadata attributes values are defined. At the granule-level, product-specific metadata is defined in the MCF. In both cases, a list of valid values and permitted ranges are specified in the ESDT data dictionary.

10.5 Testing and Validation

Testing and validation involves installation of the ESDT on a Data Server, and subsequent tests of the data services for the ESDT. These tests include insertion of actual or simulated data, search acquire, and other services that may apply and be available and supported under the extant version of ECS. (Section 11.3)

After testing, the ESDT Descriptor File and DLL are promoted to the development CM environment if pre-ECS release, or to the operational environment if after the ECS release is made operational.

This page intentionally left blank.

11. Production Rules

11.1 Introduction

Production rules are the instructions about how a particular PGE is to be run. The instructions specify a wide range of information such as the input and output data types, the frequency of execution, activation conditions and error handling instructions.

A single PGE may use one or more sets of production rules, known as PGE profiles, since it may be desirable to run the same PGE with different input granules, or activation conditions. The production rules are entered when a PGE undergoes Science Software Integration and Test (SSI&T) at the DAAC. Where applicable, default parameter values are entered at that time. The initially selected runtime parameters, metadata check parameters, and many of the default parameters can be overridden in the production environment when a Production Request is entered.

Production rules define the PGE to the Planning and Data Processing Subsystems (PDPS). The following types of conditions can be specified for each PGE:

- The time period for which the PGE will run.
  - A PGE can run every hour, every day, or for every orbit of a satellite. The frequency of how often a PGE runs must be defined to PDPS so that it knows when to plan and execute the PGE. A definition of a satellite's orbit could be included if the PGE were to be executed for some number of satellite passes.
- PGE Inputs.
  - A PGE can have any number of inputs. The types of the inputs and how frequently they are available helps determine on what basis the PGE is scheduled.
  - Most inputs to a PGE are retrieved based on time; the specified inputs are retrieved from the Data Server Subsystem for the time which the PGE is defined to execute. Production Rules allow other conditions to be added to the mix, such as checks or queries against the metadata of the input granules, or the lists of inputs as alternates (for when a primary input is not available) or optionals (for inputs without which the PGE can still run successfully). If inputs are defined as alternate or optional, the number of inputs staged for the execution of the PGE may vary from one run to the next.
- PGE Outputs.
  - A PGE can have any number of outputs. The characteristics of the outputs can have effects on any downstream PGEs that use them as inputs. For example, it is possible for an output to be defined as optional, in which case it may or may not even be produced. (When an output is not generated, it cannot be used as input for a downstream PGE.)

- Runtime Parameter Values.
  - A PGE can have any number of runtime parameters, which are values that are placed in the process control file (PCF) under specified logical IDs before the PGE executes. The PGE treats them as constants and normally they are set either during SSI&T or when the Production Request is entered.
  - For some production rules (such as Orbital Processing) there is specific information that can be placed in a runtime parameter if so desired by the PGE.
  - The following two pieces of information can be placed in a runtime parameter if so desired by the PGE:
    - Data Collection Start and Stop Times. The times are specified to the PGE via two logical IDs in the Process Control File for the PGE. Logical IDs 10258 and 10259 have the Start Data Collection and Stop Data Collection times for the current DPR run of the PGE. The date/times are only accurate to the second and may not exactly match the times in the data supplied to the PGE because PDPS can match granules that are seconds (or even minutes) from the time set for the DPR.
    - Year, Month and Day of the Data Collection Start. The year/month/day information can be placed under any Logical ID that the PGE desires. They are separate entries (each goes under a different logical ID) and can provide the PGE with the **year**, **month** and the **day** of input data collection.

Some (but not all) production rules can work with other production rules.

Production rules are often used for the selection of dynamic inputs.

- **Dynamic inputs** can be either internal or external.
  - **Dynamic internal** inputs are produced by other PGEs (they are called dynamic internal inputs because they are produced at an ECS DAAC).
  - **Dynamic external** inputs are periodically ingested and stored in the Data Server Subsystem (they are termed dynamic external inputs because they are produced outside of the DAAC).
- **Static inputs** are granules that are inserted during the SSI&T process and are retrieved not on the basis of time but by Earth Science Data Type (ESDT) and science group. Static inputs can be chosen through the following two production rules only:
  - Metadata Query.
  - Spatial Query.

PGE profiles allow a PGE to be defined to PDPS multiple times, each with a different set of inputs, outputs, or even scheduling information. Each PGE's definition is made up of its name, its version and its profile number. Different PGE name/version pairs define different PGEs to PDPS. The addition of the profile allows for multiple definitions of a PGE name/version pair. There can be up to 99 profiles for each PGE.

11.1.1 Syntax of Production Rules

Production rules are defined in the following two ways:

- Through science metadata that is entered in various types of files during the SSI&T process.
- By entering parameter values when a Production Request is created to schedule the PGE.

During SSI&T, production rules are defined in files written in Object Description Language (ODL) in a parameter equals value format. There are three general categories of ODL files:

- PGE Science Metadata ODL Files.
- ESDT Science Metadata ODL Files.
- Production Rule-Specific Science Metadata ODL Files.
  - Orbit Definition ODL Files.
  - Path Map Definition ODL Files.
  - Tile Science Metadata ODL Files.

When a Production Request is created to schedule a PGE, it is necessary to enter certain information that is essential to implementing the production rules that affect the particular PGE. The information may concern the date and time range.

11.1.2 PGE Science Metadata ODL Files

The PGE science metadata ODL file defines a PGE (or at least the current plan for its operation) to PDPS. It specifies everything from the PGE name and version, to the period for the PGE (how often it runs), all inputs and outputs, any runtime parameters and any exit messages or dependencies. A template version of the PGE science metadata ODL file is created by the **SSIT Create ODL Template** program from a PCF from the PGE.

11.1.3 ESDT Science Metadata ODL Files

The ESDT science metadata ODL file defines a PGE input or output to PDPS. Each input and output of a PGE must have a corresponding ESDT science metadata ODL file defined. It describes everything that PDPS needs to know about the subject input or output file, from its name and version, to its period (how often data is collected), to where it is used and archived. Note that many PGEs can use the same input or output type, and thus can share the same ESDT science metadata ODL file.

Unlike the PGE science metadata ODL file, there is no tool for automatically generating a template ESDT science metadata ODL file. A template version exists under the data directory called ESDT_ODL.template. The template must be copied to a file that follows the naming convention *ESDTShortName#ESDTVersionID*.odl.

11.1.4 Production Rule-Specific Science Metadata ODL Files

The production rule-specific science metadata ODL files provide specific information to PDPS about production rules used by a PGE. They are needed only when the PGE is subject to one of the following conditions:

- Is executed on the basis of a satellite orbit.
- Needs to know the orbital path of a satellite.
- Requires data based on geographic tiling of the Earth.

Since not every PGE is based on orbits, not all PGEs require these files. The comments in the PGE_ODL.template describe when setting a specific parameter means that a production rule-specific science metadata ODL file needs to be created.

The production rule-specific science metadata ODL files are broken into two types, which are defined as follows:

- Orbit ODL File.
  - Defines the orbital period of the satellite from which the PGE's input data is created.
  - Defines when a given orbit starts, how long it lasts, and the number of the orbit.
  - PDPS uses the information in the orbit ODL file to extrapolate future orbits and is able to plan PGEs that are required to run every so many orbits of the satellite.
- Pathmap ODL File.
  - Defines the mapping between the cyclic 0-233 orbits that the satellite makes with the actual path number that the PGE requires.
  - PDPS computes the path number from the orbit number (specified in the orbit ODL file) by incrementing it until it reaches the 233 maximum, then resetting it to zero.
  - Many instruments expect the path number to be a fixed swath on the Earth, so it is not just incremented for each satellite pass.
  - The Pathmap ODL file creates a mapping from the sequential path numbers to the path numbers expected by the PGEs.

Unlike the PGE science metadata ODL file, there is no tool to automatically generate a template production rule-specific science metadata ODL file. Because the files themselves tend to be small, this is not usually a problem. A template version of each kind of production rule-specific science metadata ODL file (e.g., ORBIT_ODL.template) exists in the /usr/ecs/<MODE>/CUSTOM/data directory on the AIT Workstation. The templates must be copied, named properly, and edited in order to create the appropriate production rule-specific science metadata ODL file.

11.2 Production Rules

The following statements provide some simplified descriptions of the production rules:

- **Basic Temporal** - Temporal (time) range of inputs matches the temporal range of outputs.

- **Advanced Temporal** - Temporal range of inputs is offset from the expected temporal range of inputs and outputs.
- **Alternate Inputs** - PGE is run with different inputs based on the availability of various alternate input data sets.
- **Optional Inputs** - PGE is run with specified optional inputs if available; otherwise, PGE is run without them.
- **Minimum/Maximum Number of Granules** - Minimum number of input granules needed for full data coverage and maximum number of input granules to search for may be specified. Minimum and maximum number of outputs expected from the PGE may be specified.
- **Optional DPRs** – The only DPRs executed are those for which the non-routine key input data actually become available (i.e., are either produced in data processing or can be acquired from the archive).
- **Metadata Checks** - DPR is run only if input data's metadata value(s) meet(s) certain criteria.
- **Metadata Query** - Input granule selection is based on metadata value.
- **Spatial Query/Spatial Pad** - Input granule selection is based on the spatial coverage of another input (i.e., the key input). Spatial Pad involves adding area to all sides of the key input's spatial shape. All granules that intersect the expanded area are retrieved.
- **Closest Granule** – DPR is generated if a required input granule within a particular time range (rather than an exact time) is available; otherwise, no DPR is generated. (Supersedes the Most Recent Granule Production Rule)
- **Orbital Processing** - Selection of input times is based on orbit information.
- **Multiple DPRs for Insertion Time** - Allows the creation of DPRs for multiple granules with the same insertion time (affects ASTER L1B routine processing only).
- **Tiling** - Input data is chosen on the basis of Instrument Team-defined tiles (geographic areas).

11.2.1 Basic Temporal Production Rule

The Basic Temporal Production Rule defines the timeframe for the PGE along with its input and output data. PGEs subject to the Basic Temporal Production Rule generally have the following characteristics in common:

- Typically scheduled to run using input data that become available periodically (every hour, every day, etc.).
- Use input data for a particular period of time.
- Produce output for a specified length of time.

The data the PGE takes in (its input) and the data it produces (its output) have the same period (or some subset of the same period) as the PGE.

- Example One:
  - A MODIS PGE processes data for five-minute intervals, producing Level 1B granules.

- The PGE requires as input the specific five-minute Level 1A granule that is contemporaneous with (covers the same five-minute time period as) the Level 1B granule to be produced.
- Using the Basic Temporal Production Rule, a five-minute Level 1A granule is staged as input to the PGE and a five-minute Level 1B granule is expected as output, both matching the timeframe for which the PGE is run.

- Example Two:
  - A PGE for a different instrument processes data for 24-hour intervals, producing 24-hour Level 1A granules as output.
  - As input the PGE takes Level 0 data that is ingested every two hours.
  - Using the Basic Temporal Production Rule, twelve two-hour Level 0 granules are staged as input to the PGE and a 24-hour Level 1A granule is expected as output, matching the timeframe for which the PGE is run.

The fundamental elements used to define the Basic Temporal Production Rule are "period" and "boundary."

- **Period** is the length of time for which a PGE processes data or the length of time for which input and output data is collected.
  - A PGE that is subject to the Basic Temporal Production Rule only and that processes data in two-hour blocks, takes in data that relates to a particular two-hour interval and produces output data for that same two-hour period.
  - Data that has a period of 15 minutes was collected or produced for a 15-minute time period.
- **Boundary** is the starting point for the data or PGE.
  - Depending on the characteristics of the data or PGE, the boundary may be the start of a minute or hour or day or week (etc.).
  - If a PGE's boundary is the start of the hour, it processes data that starts every hour and runs on data for the length of its period.
  - If data comes in every day, PDPS predicts that the data is going to be available at the start of the day and allows scheduling of PGEs that use the data as input accordingly.

Both the PGE itself and the input data have a boundary and period associated with them. That is how PDPS determines the frequency of processing for a Basic Temporal PGE and the time period for its inputs and outputs.

PDPS uses **period** and **boundary** in combination to plan the processing of each PGE, including determining its input requirements and anticipated output (which may be input to other PGEs). If a PGE has a period of one hour and a boundary of "start of day," it is scheduled every hour, beginning at midnight. If an input has a period of 15 minutes and boundary of "start of hour," PDPS predicts it every 15 minutes beginning on the hour.

Boundary offset is an addition to the Basic Temporal Production Rule that allows a PGE or data to start on an offset from a given boundary. For example, if a PGE would normally run every day but not start until two or three hours into the day (e.g., beginning at 3:00 a.m. instead of midnight), a boundary offset can be used to add three hours to the “start of day” boundary. This would mean the PGE would run on data that occurred three hours after the boundary.

Non-routine data can be used as an input to a Basic Temporal DPR. Such data must exist at the Science Data Server for the time period of the DPR when the DPR is created or it can be retrieved using the **Closest Granule** or **Optional Input** Production Rules. If no other Production Rule is used and the non-routine data is **not** available at Science Data Server, the DPR fails at creation time. Because there is no Boundary or Period associated with non-routine data, PDPS cannot predict it; consequently, it must be available at Production Request time.

Running with data that falls within the time of the DPR only is a special case that is applicable to some PGEs. Normally, PDPS would match any granules that intersect the time of the DPR. However, some PGEs require data that falls within the time of the DPR only. For example:

- A data set is composed of granules with the following time ranges:
  - 01:00-02:00
  - 02:00-03:00
  - 03:00-04:00
- The DPR has the following time range:
  - 01:30-03:30

Usually a PGE would want all three granules in the data set because they all intersect the time of the DPR. But if a PGE wants the data that falls within the DPR time only, the 02:00-03:00 granule would be the only one selected. Restriction of the data to the time of the DPR is accomplished using the **DPR Align Flag** (ALIGN_DPR_TIME_WITH_INPUT_TIME).

The **end-of-month anomaly** is an addition to the Basic Temporal Production Rule that allows a PGE or data to cover a specific number of days within a month. The month is broken into thirds. The first third is composed of the first 10 days of the month. The second third consists of days 11 through 20. And the last third varies in length depending on the total number of days in the month (i.e., for November it would have 10 days; for December it would have 11 days). A specific **boundary** and **period** allow a PGE or its data to be scheduled into thirds of a month.

Figure 11.2.1-1 provides an illustration of the Basic Temporal Production Rule. The PGE has a boundary of “start of day” and a period of one hour, so it is scheduled for every hour through the day. If a Production Request were entered for two full days of processing, a DPR would be created for the PGE to run every hour; i.e., 48 DPRs total. If a Production Request were created for a four-hour period in the middle of a single day (for example, from 12:00 noon to 4:00 p.m.), then four DPRs would be created, one for 12:00-1:00, one for 1:00-2:00, one for 2:00-3:00, and one for 3:00-4:00.

In the example (Figure 11.2.1-1), Input One has a boundary of “start of day” and a period of two hours, so when PDPS plans for its availability, it expects a granule every two hours beginning at midnight. Consequently, each granule of Input One is associated with two DPRs for the PGE, because the PGE encompasses only one hour of the two-hour granule's period.

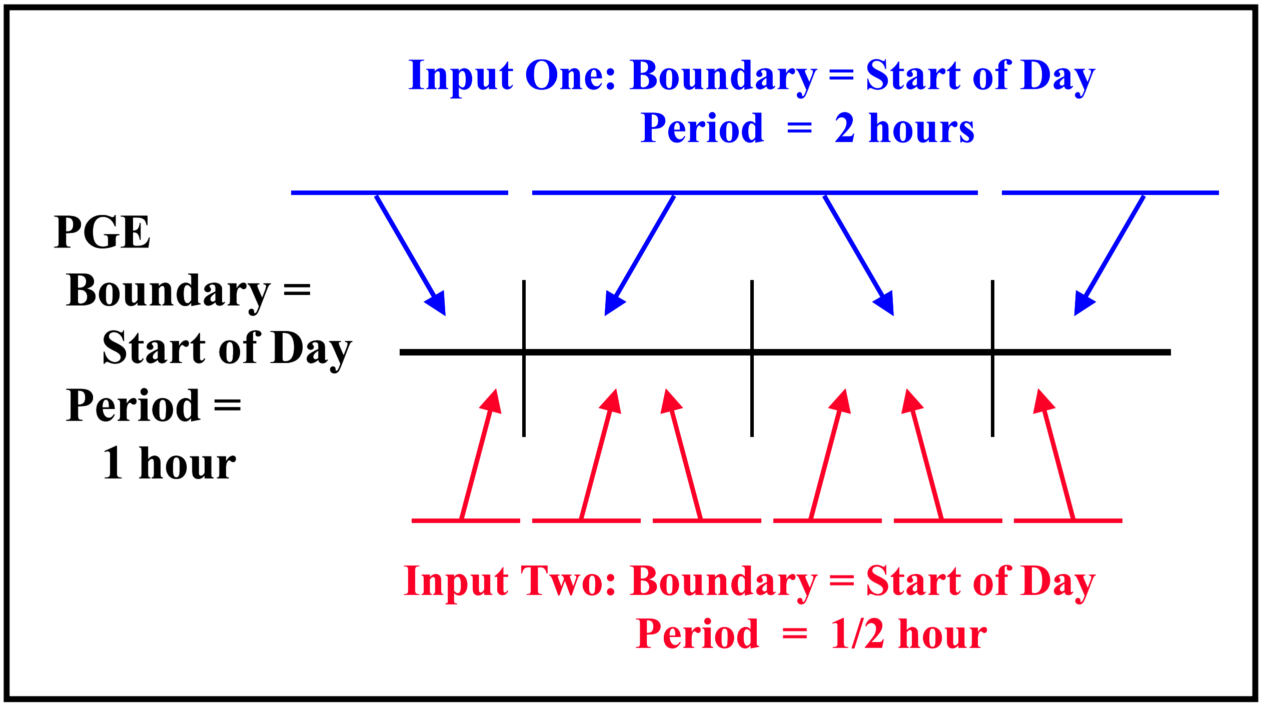

Figure 11.2.1-1. Example of the Basic Temporal Production Rule

Input Two has a boundary of “start of day” and a period of ½ hour, so when PDPS plans for its availability, it expects a granule every ½ hour beginning at midnight. As a result two granules of Input Two are associated with each DPR for the PGE, because the PGE encompasses an hour of the ½-hour granule's Period. Thus, every DPR of the PGE will wait for two granules of Input Two to arrive before it can be processed.

11.2.1.1 PGE Science Metadata ODL File Parameters

The following parameters must be set properly in the applicable PGE science metadata ODL file in order to implement the Basic Temporal Production Rule:

- SCHEDULE_TYPE.
- PROCESSING_PERIOD.
- PROCESSING_BOUNDARY.

The SCHEDULE_TYPE parameter specifies the type of scheduling that will be done for the PGE. The following values are applicable to the Basic Temporal Production Rule:

- "Time"
  - The PGE is scheduled on the basis of the specified boundary/period and the availability of data for that boundary/period.
- "Snapshot"
  - The PGE is scheduled for a single date/time.
  - Note that PROCESSING_PERIOD and PROCESSING_BOUNDARY are not needed when "Snapshot" is specified.

Other values for SCHEDULE_TYPE apply to other production rules, such as the following values:

- “Data"
  - The PGE is scheduled on the basis of the availability of data produced by other PGEs.
- "Tile"
  - The PGE is scheduled based on the definition of geographic tiles.
- "Orbit"
  - PGE scheduling is based on the orbit of the spacecraft.

The PROCESSING_PERIOD parameter describes the length of time for which the PGE executes. Data will be acquired (barring any combination of Production Rules) for the specified period and output data will be planned for the given period. It is of the format "<Period Type>=<Length of Period>". Note that “length of period” can be specified as a positive integer only. The following values are acceptable “period type” entries for the Basic Temporal Production Rule:

- "YEARS"
  - PGE processes data applicable to a given year or years.
  - "YEARS" might be specified for a PGE that computes a yearly average.
  - For example, PROCESSING_PERIOD = "YEARS=1" relates to a PGE that processes one year’s worth of data.
- "MONTHS"
  - PGE processes data applicable to a particular month or several months.
  - “MONTHS” is most likely to be used for some kind of averaging PGE.
  - For example, PROCESSING_PERIOD = "MONTHS=2" relates to a PGE that processes two months’ worth of data at a time.
- "THIRDS"
  - PGE processes data applicable to some number of thirds of the month.
  - For example, PROCESSING_PERIOD = "THIRDS=1" relates to a PGE that processes data applicable to 1/3 of the month.
- "WEEKS"
  - PGE processes data applicable to some number of weeks.
  - For example, PROCESSING_PERIOD = "WEEKS=2" relates to a PGE that processes two weeks’ worth of data every time it runs.
- "DAYS"
  - PGE processes data applicable to some number of days.
  - For example, PROCESSING_PERIOD = "DAYS=5" relates to a PGE that processes five days’ worth of data.
- "HOURS"
  - PGE processes data applicable to some number of hours.
  - For example, PROCESSING_PERIOD = "HOURS=4" relates to a PGE that processes four hours’ worth of data when it is executed.

- "MINS"
  - PGE processes data applicable to some number of minutes.
  - For example, PROCESSING_PERIOD = "MINS=5" relates to a PGE that processes five minutes' worth of data.
- "SECS"
  - PGE processes data applicable to some number of seconds.
  - For example, PROCESSING_PERIOD = "SECS=2" relates to a PGE that runs on two seconds' worth of data.

There are other types of values for PROCESSING_PERIOD but they apply to other production rules (as described in the applicable sections of the lesson).

The PROCESSING_BOUNDARY parameter specifies the boundary (starting point in time) of the PGE. It tells when each instance of the PGE should start. Note that the PROCESSING_BOUNDARY and PROCESSING_PERIOD are used in conjunction to schedule the PGE.

The following PROCESSING_BOUNDARY values are used for implementing the Basic Temporal Production Rule:

- "START_OF_HOUR" – PGE processes data for each hourly interval.
- "START_OF_6HOUR" - PGE processes data for every 6-hour interval.
- "START_OF_DAY" - PGE processes data for every daily interval.
- "START_OF_WEEK" - PGE processes data for every weekly interval.
- "START_OF_ONE_THIRD_MONTH" - PGE processes data for every 1/3 of a month.
- "START_OF_MONTH" - PGE processes data for every monthly interval.
- "START_OF_YEAR" - PGE processes data for every yearly interval.
- "START_OF_TIME" - PGE runs from the specified date/time in the form "START_OF_TIME=DD/MM/YYYY HH:MM:SS".
  - Can be used in conjunction with all types of PROCESSING_PERIOD.

There are other values for PROCESSING_BOUNDARY that apply to other production rules (as described in the applicable sections of the lesson).

11.2.1.2 Retrieving Data that Falls within the DPR Time Only

When the ALIGN_DPR_TIME_WITH_INPUT_TIME flag is set to "Y" (i.e., ALIGN_DPR_TIME_WITH_INPUT_TIME = "Y") PDPS uses only the granules for the input that fall within (instead of intersecting) the time of the DPR. If the flag is **not** set or is set to "N", granules for the input will be chosen on the basis of whether or not they intersect the time of the DPR.

11.2.1.3 ESDT Science Metadata ODL File Parameters

The following parameters must be set properly in the applicable ESDT science metadata ODL file in order to implement the Basic Temporal Production Rule:

- DYNAMIC_FLAG.
- PERIOD.
- BOUNDARY.

The DYNAMIC_FLAG describes the type of data that is defined in the ESDT science metadata ODL file. It specifies to PDPS what kind of data the PGE requires as input or produces as output. It can have any of the following four possible values, all of which are valid for Basic Temporal data:

- "S"
  - Static Data.
  - Data do not change at regular intervals.
  - The same granule can be used as input for many runs of the PGE.
  - Calibration files are a good example of static data.
- "I"
  - Dynamic Internal.
  - Data are produced by a PGE running at the local DAAC.
  - All output products are either "dynamic internal" or "interim" kinds of data.
- "E"
  - Dynamic External.
  - Data are produced by an external source (not a PGE running at the local DAAC).
  - Earth Observing System (EOS) Data and Operations System (EDOS) data is a primary example.
  - Dynamic external can be set for PGE inputs only.
- "T"
  - Interim/Intermediate.
  - Data are stored only temporarily by the Data Server Subsystem.

The PERIOD parameter specifies the length of time covered by the data. Data are expected to be either ingested or produced for the length of the PROCESSING_PERIOD described in PGE science metadata ODL files. However, the PERIOD of the data does **not** have to match the PROCESSING_PERIOD defined for the PGE. PDPS plans for data where the ESDT period is less or more than the processing period of the PGE that uses it. For example, if the PGE PROCESSING_PERIOD = "HOURS=1" and the input data PERIOD = "MINS=5", then PDPS plans to acquire twelve granules of the input data to cover the PROCESSING_PERIOD.

The following "period type" values are used for implementing the Basic Temporal Production Rule:

- "YEARS"
  - Data span a year or years.
  - "YEARS" might be selected for a yearly average output product.
  - For example, PERIOD = "YEARS=1" specifies data that cover a period of a year.
- "MONTHS"
  - Data span a month or several months.
  - "MONTHS" is most likely used for some kind of averaging output product.

- For example, PERIOD = "MONTHS=2" specifies data that cover a period of two months.
- "THIRDS"
  - Data span some number of thirds of a month.
  - For example, PERIOD = "THIRDS=1" specifies data that cover a period of 1/3 month.
- "WEEKS"
  - Data span some number of weeks.
  - For example, PERIOD = "WEEKS=2" specifies data that cover a period of two weeks.
- "DAYS"
  - Data span some number of days.
  - For example, PERIOD = "DAYS=5" specifies data that cover a period of five days.
- "HOURS"
  - Data span some number of hours.
  - For example, PERIOD = "HOURS=4" specifies data that cover a period of four hours.
- "MINS"
  - Data span some number of minutes.
  - For example, PERIOD = "MINS=5" specifies data that cover a period of five minutes.
- "SECS"
  - Data span some number of seconds.
  - For example, PERIOD = "SECS=2" specifies data that cover a period of two seconds.
- "ORBITS"
  - Data span some number of orbits of the spacecraft.
  - For example, PERIOD = "ORBITS=1" specifies data that cover one orbit.
  - A PGE can be time-scheduled (using the Basic Temporal Production Rule) but use orbit-based data.

The BOUNDARY parameter is the starting point in time of the data granule. It tells when each data granule should start. Note that the BOUNDARY and PERIOD are used in conjunction to determine the starting and ending time for the granules.

The following values for BOUNDARY apply to the Basic Temporal Production Rule:

- "START_OF_HOUR"
  - Data granules start every hour.
- "START_OF_6HOUR"
  - Data granules start every six hours.
- "START_OF_DAY"
  - Data granules start every day.

- "START_OF_WEEK"
  - Data granules start every week.
- "START_OF_ONE_THIRD_MONTH"
  - Data granules start every 1/3 of a month.
- "START_OF_MONTH"
  - Data granules start every month.
- "START_OF_YEAR"
  - Data granules start every year.
- "START_OF_ORBIT"
  - Data granules start every orbit.

The PREDICTION_METHOD parameter specifies the method of prediction that the Planning and Data Processing Systems use when creating DPRs that use the ESDT. The parameter has two possible values:

- "ROUTINE"
  - Data granules come in at regular intervals.
  - Boundary and Period are used when predicting the inputs to DPRs.
- "NONROUTINE"
  - Data granules come in at odd or random intervals.
  - DPRs that use the input can work off data that exists in the Science Data Server when the production request is created. This is because PDPS cannot predict the data.
  - When PREDICTION_METHOD = NONROUTINE, Boundary and Period are **not** used.

11.2.2 Advanced Temporal Production Rule

The Advanced Temporal Production Rule allows for input data to be acquired for a time period other than that of the PGE or its planned inputs/outputs. It provides an offset mechanism, specifying on an input basis that the data required for processing is some number of seconds earlier or later than the planned time period for the PGE.

- Example One:
  - A PGE requires data from its previous execution for interpolation purposes (e.g., one of its inputs is the output of the very same PGE the last time that it ran).
  - If the PGE processes data for each one-hour interval (producing an hourly product), the Advanced Temporal Production Rule is specified with an offset of minus 3600 seconds (one hour) for the input of the ESDT produced by previous runs.
- Example Two:
  - A PGE takes as input two-hour Level 0 data to produce an L1A product.
  - Because the edges of the Level 0 data can be difficult to process without preceding and succeeding data, the PGE requires three Level 0 granules, one from the time period before it runs, one for the time period it is currently processing and one for the next time period.

- The PGE is defined as having three inputs, the first with an Advanced Temporal offset of minus 7200 seconds (two hours), the second with no Advanced Temporal offset and the third with an Advanced Temporal offset of plus 7200 seconds (two hours).

The Advanced Temporal Production Rule uses the times specified in the Basic Temporal Production Rule as a reference point for specifying offset(s) to request data from a "period" and/or "boundary" different from that of the DPR or its input. The offsets are specified as either negative or positive numbers to indicate whether the time period of the input data is before or after that of the DPR (a particular run of a PGE).

- **Begin Period Offset** is an amount of time (in seconds) that is specified with respect to the DPR start time. A negative beginning offset requests data that was collected before the DPR start time. A positive beginning offset requests data with a collection time after the start time of the DPR.
- **End Period Offset** is an amount of time (in seconds) that is specified with respect to the DPR end time. A negative ending offset requests data that ended collection before the DPR end time was reached. A positive ending offset requests data that ended collection after the end time of the DPR boundaries.

Note that the beginning and ending offsets are not absolute cut-offs for data. Overlapping granules (granules that start or end outside of the offsets) will be staged as inputs to the DPR.

Figure 11.2.2-1 provides an illustration of the Advanced Temporal Production Rule. The PGE shown in the example processes data for every one-hour interval. However, Input One comes in at two-hour intervals and Input Two is produced every 1/2 hour.

Both the Begin Period Offset and End Period Offset for Input One are -7200 seconds (minus two hours). Consequently, every DPR will stage the "previous" Input One. This could be used to get the "previous" or "next" granule of an input.

The Begin Period Offset for Input Two is zero, meaning that it will match the Start Time of the DPR. The End Period Offset is +1800 seconds (plus 1/2 hour). Therefore, all Input Two granules that fall within the time period of the DPR plus 1/2 hour would be staged. The effect is to acquire all Input Two granules within the time period of the DPR, plus the one from the next 1/2-hour time period, for a total of three granules. The additional granule acquired by means of the End Period Offset might be used for interpolation purposes at the end point.

The same types of parameter settings that apply to the Basic Temporal Production Rule apply to the Advanced Temporal Production Rule. In addition, there are some parameters in the PGE science metadata ODL file that apply specifically to the Advanced Temporal Production Rule. However, the values applicable to the Basic Temporal Production Rule must be set before the Advanced Temporal Production Rule syntax is added.

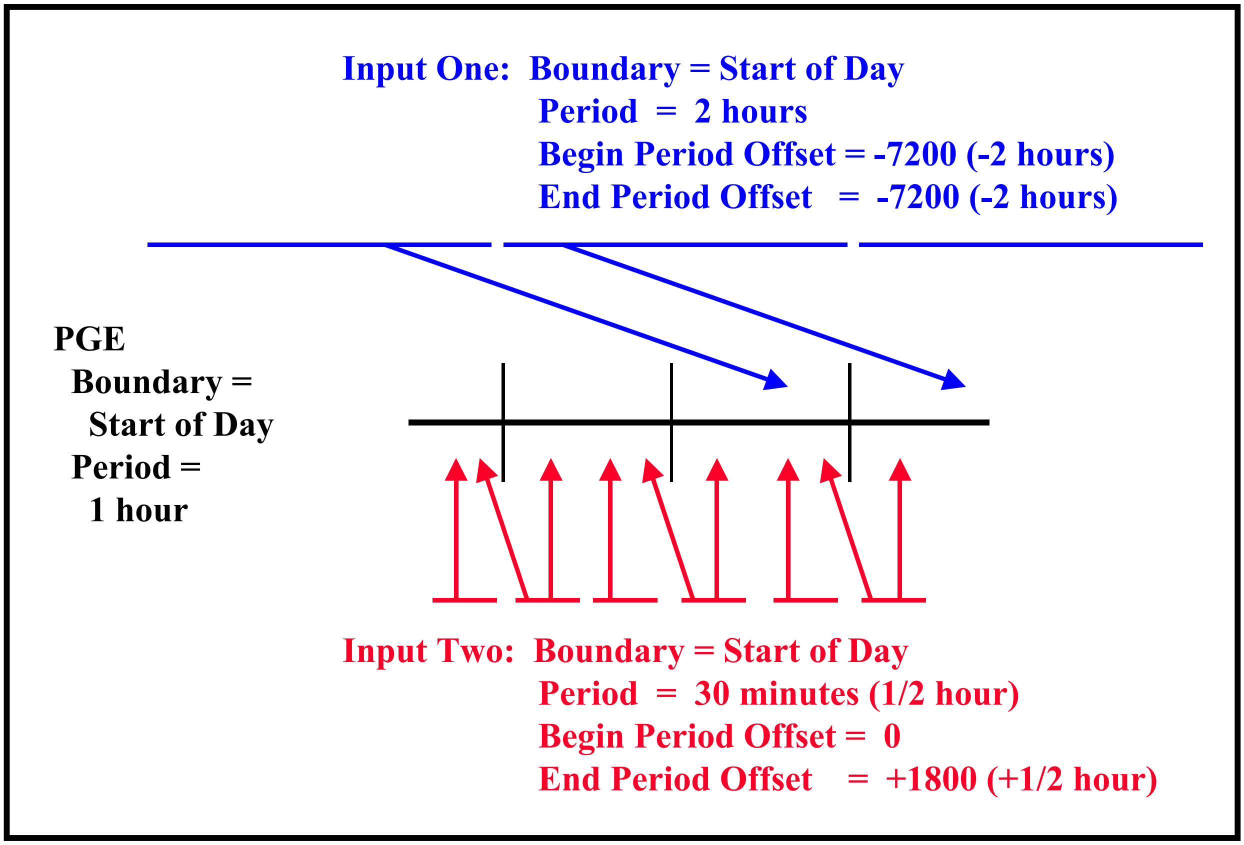

Figure 11.2.2-1. Example of the Advanced Temporal Production Rule

11.2.2.1 PGE Science Metadata ODL File Parameters

During the SSI&T process the PGE science metadata ODL file is generated from the PCF delivered with the science algorithm. A PCF_ENTRY object is generated for each file entry in the PCF. In order to implement the Advanced Temporal Production Rule the PCF_ENTRY object for each type of input file to which the rule applies uses the following syntax:

```
OBJECT = PCF_ENTRY
  .
  .
  .
  BEGIN_PERIOD_OFFSET =
  END_PERIOD_OFFSET =
  .
  .
  .
END_OBJECT = PCF_ENTRY
```

Accordingly, the following parameters must be set properly in order to implement the Advanced Temporal Production Rule:

- BEGIN_PERIOD_OFFSET.
- END_PERIOD_OFFSET.

BEGIN_PERIOD_OFFSET is the offset added to or subtracted from the Data Start Time of the DPR. The value assigned to BEGIN_PERIOD_OFFSET can be either a positive or negative value, specified in seconds. If the value is positive, it is added to the Data Collection Start Time (looking for the input forward in time). If the value is negative, it is subtracted from the Data Collection Start Time (looking backward in time). For example, BEGIN_PERIOD_OFFSET = -3600 requests data that was collected one hour (3600 seconds) before the DPR start time.

END_PERIOD_OFFSET is the offset added to or subtracted from the Data Collection End Time of the DPR. The value assigned to END_PERIOD_OFFSET can be either a positive or negative value, specified in seconds. If the value is positive, it is added to the Data Collection End Time (looking for the input forward in time). If the value is negative, it is subtracted from the Data Collection End Time (looking backward in time). For example, END_PERIOD_OFFSET = +2700 requests data that was collected 45 minutes (2700 seconds) after the DPR end time.

The BEGIN_PERIOD_OFFSET and END_PERIOD_OFFSET parameters can be specified for any input PCF_ENTRY in the PGE science metadata ODL file. If not specified, the parameters are set to zero (0) and the Advanced Temporal Production Rule does not apply to the PGE.

11.2.3 Alternate Input and Optional Input Production Rules

The Alternate Input and Optional Input Production Rules are very similar and use much the same processing in PDPS. Both rules allow a PGE to select various inputs based on timers and priority lists. The major difference is that Alternate Inputs requires that one of alternates on the list be used, whereas Optional Inputs allows successful execution of the PGE if no optional input on the list is available.

The Alternate Input Production Rule allows for a PGE to evaluate a list of inputs in priority order and be scheduled and executed with the best priority input that could be found. In essence, a PGE using Alternate Inputs is saying "I would like to run with Input A, but if it's not available, I am willing to run with Input B." A timer can be used to specify how long to wait for a given alternate choice before proceeding with a choice of lesser priority. The PGE is not executed until one of the alternate choices has been found.

- Example:
  - A PGE requires model wind data as an input but is capable of accepting wind data from a Data Assimilation Office (DAO) model, a National Centers for Environmental Prediction (NCEP) model, or (as a last resort) climatology.
  - The PGE would use the Alternate Input Production Rule to list each input in priority order, giving a timer value for how long to wait before trying the next input.
  - If the DAO data were most desirable, DAO would be listed as first choice or "primary" data.
  - NCEP would be the second choice.
  - Climatology would be the last choice.

- If a timer value is specified for DAO data, the PGE will wait for that timer to expire before running with either NCEP data or climatology.
- If a timer had been placed on the NCEP input, the PGE would wait before running with the climatology data.

The Optional Input Production Rule allows for a PGE to list inputs that are desired but not required for it to execute. The inputs are ranked as previously stated and timers are set to wait before choosing a lower-priority type of input. However, if none of the inputs on the list becomes available, the PGE starts because the alternatives are classified as "optional." In essence the PGE is saying "I would like to run with Input A, but if its not available, I can run (and produce reasonable output) without it."

- Example:
  - It would be preferable to run a particular MODIS PGE with the output of a MISR PGE as input.
  - However, the MISR output may not be produced every day.
  - So the MODIS PGE lists the MISR input as optional with a two-hour timer.
  - On those occasions when no MISR output is produced, the MODIS PGE waits for two hours and then is executed without the MISR input.

Figure 11.2.3-1 provides an illustration of the Alternate Input Production Rule. The PGE in the illustration has two inputs that are “required” so they must available for the PGE to be run. It also has one input that is “alternate.” The alternate input can be one of three choices, the first choice is the **primary**, then there are second and third choices.

After the pair of required inputs has become available, the alternate inputs are evaluated as follows:

- If the primary alternate is available, it is used as input and the PGE is scheduled for execution.
- There is a one-hour timer on the primary alternate. If the primary alternate is unavailable, the PGE waits until the primary alternate becomes available or the one-hour timer expires, whichever occurs first.
- If the second alternate is available after the timer for the primary alternate has expired, the second alternate is used as input and the PGE is scheduled for execution.
- There is a four-hour timer on the second alternate. If the second alternate is unavailable, the PGE waits until either the primary alternate or the secondary alternate becomes available or the four-hour timer expires, whichever occurs first.
- If the third alternate is available after the timer for the second alternate has expired, the third alternate is used and the PGE is scheduled for execution.
- There is no timer on the third alternate. If the third alternate is not available, the PGE waits until either the primary alternate, the secondary alternate, or the third alternate becomes available, whichever occurs first.
- The PGE will not start processing until one of the alternates becomes available.

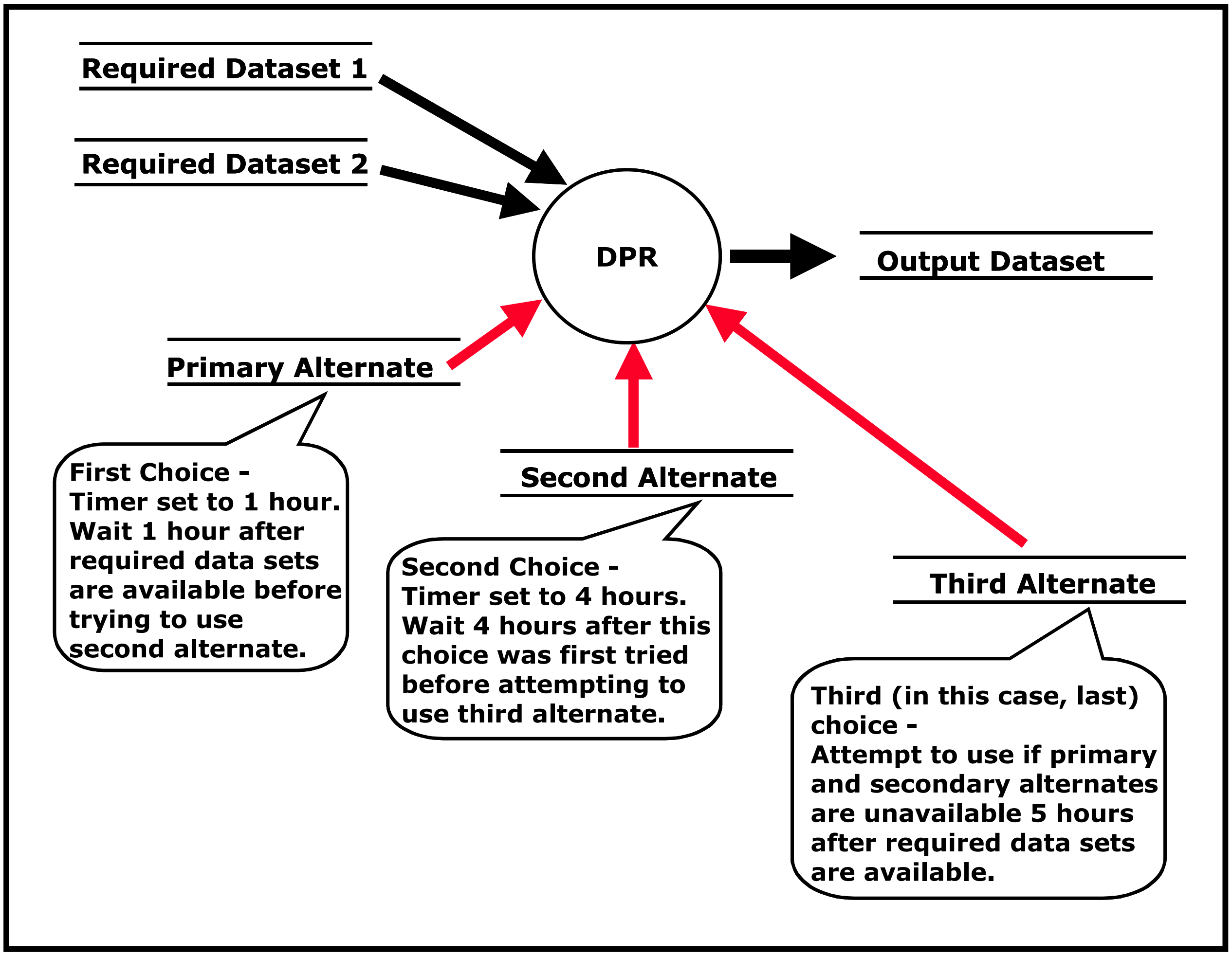

Figure 11.2.3-1. Example of the Alternate Input Production Rule

If instead of an alternate the third input for the PGE had been defined as an optional input, the preceding scenario would have been the same, except that if neither the primary alternate, the second alternate nor the third option was available after the timers had expired, the PGE would not wait; it would be scheduled for execution without the third input. It would run with the two required inputs only.

The Alternate Input and Optional Input Production Rules are additions to settings/syntax put into the ODL files for other production rules. Inputs deemed "optional" or "alternate" can be searched for and acquired by other production rules (e.g., Basic Temporal or Metadata Checks/Query). The syntax for the rules used to search for the inputs have to be filled out in addition to the syntax required to make the input an alternate or optional input.

11.2.3.1 PGE Science Metadata ODL File Parameters

The following parameter must be set properly in the applicable PGE science metadata ODL file in order to implement the Alternate Input or Optional Input Production Rule:

- INPUT_TYPE.

In addition, one of the following two ODL objects is used within a PCF_ENTRY to define either the Alternate Input Production Rule or the Optional Input Production Rule:

- ALTERNATE_INPUT object.
- OPTIONAL_INPUT object.

INPUT_TYPE is a type of data defined by a PCF_ENTRY object (i.e., between OBJECT = PCF_ENTRY and END_OBJECT = PCF_ENTRY). It can have one of four possible values, only three of which are used to define an alternate or optional input:

- "Required"
  - A required input.
  - The data must be available or the PGE does not execute.
  - It is the "normal" value for the parameter (i.e., INPUT_TYPE = “Required”); consequently, the input is neither an alternate input nor an optional input.
- "Primary"
  - The primary alternate input.
  - The data is the first choice in a list of alternates.
- "Alternate"
  - An alternate input (except the primary alternate) in a list of alternates.
  - The data is not the first choice in a list of alternates; it is a subsequent choice if the primary (or a higher-priority alternate) is not available.
- "Optional"
  - An optional input.
  - Availability of the data will be checked and if a timer has been specified, execution of the PGE will wait.
  - The PGE can be executed without the data if it is not available.

Although the Alternate Input and Optional Input Production Rules are similar, there are two different ODL objects used to define them within a PCF_ENTRY; i.e., the ALTERNATE_INPUT object and the OPTIONAL_INPUT object.

The ALTERNATE_INPUT object has the following syntax:

```
OBJECT = PCF_ENTRY
  .
  .
  .
  OBJECT = ALTERNATE_INPUT
    .
    .
    .
  END_OBJECT = ALTERNATE_INPUT
END_OBJECT = PCF_ENTRY
```

The ALTERNATE_INPUT ODL object surrounds an Alternate Input definition. An OBJECT/END_OBJECT pair separates the parameters defining the Alternate Input from the rest of the parameters defining the PCF_ENTRY. The following parameters define an ALTERNATE_INPUT object:

- CLASS.
- CATEGORY.
- ORDER.
- RUNTIME_PARM_ID.
- TIMER.
- WAITFOR.
- TEMPORAL [not implemented].

CLASS is a simple counter used to differentiate the different ALTERNATE_INPUT objects within the file. Since each ALTERNATE_INPUT object resides within a different PCF_ENTRY object, the CLASS for an ALTERNATE_INPUT object can always be 1.

CATEGORY is the name of the list of alternates to which the ALTERNATE_INPUT belongs. The PDPS uses CATEGORY to associate different alternates within a list. CATEGORY can be set to any string value of 20 characters or less (e.g., CATEGORY = "Snow Ice"). Alternates that are part of the same list should have matching CATEGORY values.

ORDER is the numerical place that the particular alternate holds in the list of alternates. The first choice or Primary Alternate (with the INPUT_TYPE = "Primary") should have ORDER = 1.

RUNTIME_PARM_ID specifies the Logical ID (in the PCF) for which the PGE will find the Logical ID of the alternate chosen. Since all alternates must be contained within different PCF_ENTRY objects, they all must have different Logical IDs (but all alternates within the same CATEGORY should have the same value of RUNTIME_PARM_ID). The RUNTIME_PARM_ID parameter specifies the Logical ID of a runtime parameter that the PGE may read to find out which alternate was chosen for the particular execution of the PGE.

The TIMER parameter specifies how long to wait for the particular alternate before checking for the next alternate in the list. The parameter value is expressed in the format "<Period Type>=<Length of Period>". Note that “Length of Period” can be specified as a positive integer only.

The Alternate Input Production Rule accepts the following “Period Type” values:

- "WEEKS"
  - PDPS should wait for some number of weeks before searching for the next alternate in the list.
  - For example, TIMER = "WEEKS=2" would make PDPS wait two weeks before checking for the next alternate input.
- "DAYS"
  - PDPS should wait for some number of days before searching for the next alternate in the list.

- For example, TIMER = "DAYS=5" would make PDPS wait five days before checking for the next alternate input.
- "HOURS"
  - PDPS should wait for some number of hours before searching for the next alternate in the list.
  - For example, TIMER ="HOURS=4" would make PDPS wait four hours before checking for the next alternate input.
- "MINS"
  - PDPS should wait for some number of minutes before searching for the next alternate in the list.
  - For example, TIMER = "MINS=5" would make PDPS wait five minutes before checking for the next alternate input.
- "SECS"
  - PDPS should wait for some number of seconds before searching for the next alternate in the list.
  - For example, TIMER = "SECS=2" would make PDPS wait two seconds before checking for the next alternate input.

The WAITFOR parameter specifies whether or not the PGE can be run without the alternate input. Setting WAITFOR = "N" means that the PGE can run without the input if it cannot be found. In a list of alternate inputs, this would have meaning for the last choice only. If WAITFOR = "Y", the PGE is not executed (even after the last alternate timer expires) until one of the alternates in the list can be found.

The TEMPORAL parameter is an unimplemented feature that would allow for searching for alternates from the same time period but a different date. It is currently stored in the PDPS database but is not used.

The OPTIONAL_INPUT object has the following syntax:

```
OBJECT = PCF_ENTRY
  .
  .
  .
  .
  .
  OBJECT = OPTIONAL_INPUT
    .
    .
    .
  END_OBJECT = OPTIONAL_INPUT
END_OBJECT = PCF_ENTRY
```

The OPTIONAL_INPUT ODL object surrounds an Optional Input definition. An OBJECT/END_OBJECT pair separates the parameters defining the Optional Input from the rest of the parameters defining the PCF_ENTRY. The following parameters define an OPTIONAL_INPUT object:

- CLASS.
- CATEGORY.
- ORDER.
- RUNTIME_PARM_ID.
- TIMER.
- TEMPORAL [not implemented].

The parameters that apply to the Optional Input Production Rule are defined in the same way that the corresponding parameters are defined for the Alternate Input Production Rule. However, note that the Optional Input Production Rule has no WAITFOR parameter. It is irrelevant; in fact, the very essence of the Optional Input Production Rule depends on not "waiting for" the last option but going ahead with the execution of the PGE without the unavailable optional input(s).

11.2.4 Minimum/Maximum Number of Granules Production Rule

The Minimum/Maximum Number of Granules Production Rule makes it possible to specify a range of possible granules for a given input or output for a PGE.

- Inputs.
  - Minimum number of granules the PGE needs for full data coverage.
  - Maximum number of granules for the time period.
- Outputs.
  - Minimum number of outputs that the PGE is expected to produce.
  - Maximum number of outputs that the PGE is expected to produce.

For example, a PGE processes data for every 90-minute interval, has a period of 90 minutes, and takes as input a granule with a period of two hours.

- In many instances one granule of the input will satisfy the PGE.
- In other instances, because of the way the two-hour and 90-minute periods overlap, the PGE needs two input granules to cover the time period.
- Therefore:
  - Minimum Number of Granules = 1.
  - Maximum Number of Granules = 2.

The Minimum/Maximum Number of Granules Production Rule is different from most production rules because it works for both input and output granules. It allows the PGE to request of a range of inputs (i.e., 1-10 granules), so that it runs with as few as one granule but with as many as ten granules. If a PGE needs at least three granules of a particular input, the minimum number of granules is defined as three and the PGE is not executed until at least three granules are available.

Optional outputs are defined when the Minimum Number of Granules is set to zero. In such cases the PGE can produce none of the particular type of output and still be considered to have executed successfully. If a PGE has a non-zero value for a Minimum Number of Granules associated with an output, and fails to produce any granules of that output type, it is marked as failed.

Figure 11.2.4-1 provides an illustration of the Minimum/Maximum Number of Granules Production Rule. In the example the PGE processes data related to a one-hour period and takes in both Input 1 and Input 2. Since Input 1 has a PERIOD of 1/2 hour, every PGE run requires two Input 1 granules. Input 2 has a PERIOD of 15 minutes, so there are four Input 2 granules for every PGE run.

The PGE produces three Output 1 granules for each run. In this case it does not produce any Output 2 granules.

Minimum and maximum values can affect each input and output as follows:

- Input 1:
  - If Minimum Granules is set to anything equal to or less than two for Input 1, the PGE is scheduled and executed.
  - If Minimum Granules is set to three, the PGE is not scheduled because there are not enough Input 1 granules to make the minimum.
  - If Maximum Granules is set to anything equal to or greater than two for Input 1, the PGE is scheduled and executed.
  - If Maximum Granules is set to one, the PGE is not scheduled because there are too many Input 1 granules (the number exceeds the maximum that the PGE can process).
- Input 2:
  - If Minimum Granules is set to anything equal to or less than four for Input 2, the PGE is scheduled and executed. If Minimum Granules is set to five, the PGE is not scheduled because there are not enough Input 2 granules to make the minimum.
  - If Maximum Granules is set to anything equal to or greater than four for Input 2, the PGE is scheduled and executed.
  - If Maximum Granules is set to three, the PGE is not scheduled because there are too many Input 2 granules (the number exceeds the maximum that the PGE can process).
- Output 1:
  - If Minimum Granules is set to anything equal to or less than three for Output 1, the PGE is scheduled and executes successfully.
  - If Minimum Granules is set to four, the PGE is marked as failed because it did not produce the expected number of output granules.
  - If Maximum Granules is set to anything equal to or greater than three for Output 1, the PGE is scheduled and executes successfully.
  - If Maximum Granules is set to two, the PGE is marked as failed because it produced too many output granules.

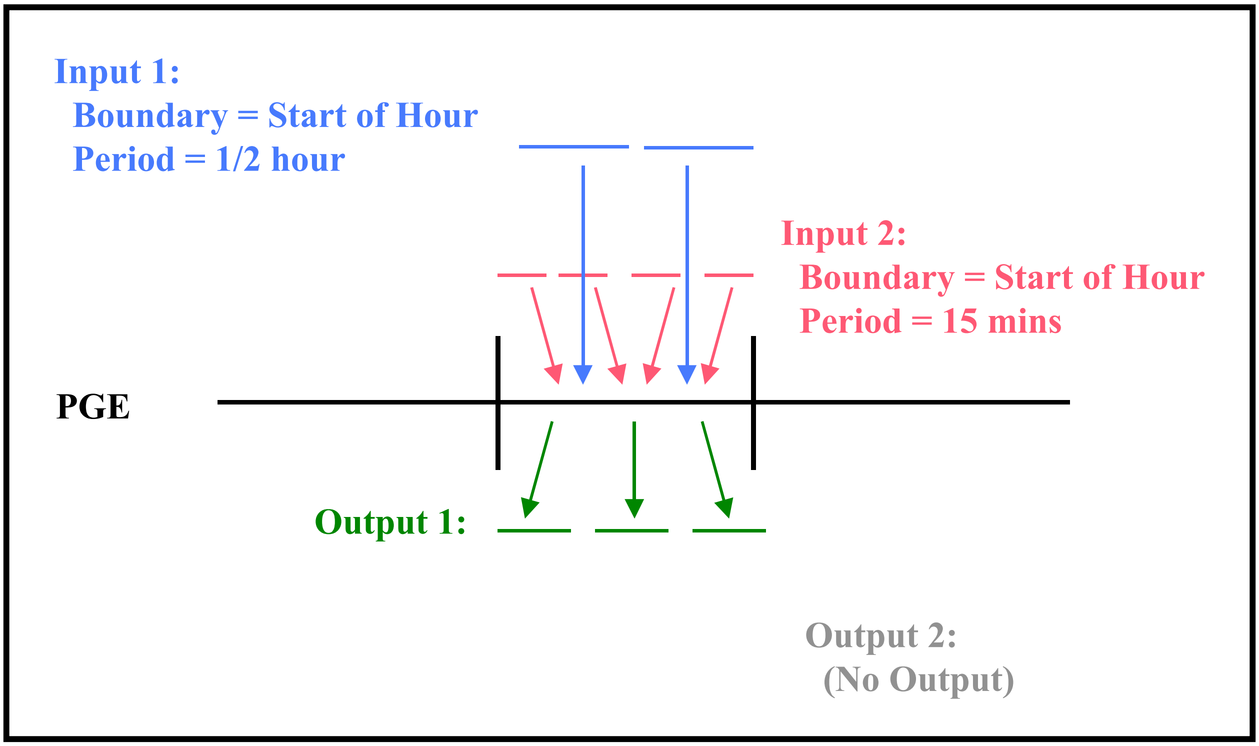

Figure 11.2.4-1. Example of the Minimum/Maximum Number of Granules Production Rule

- Output 2:
  - If Minimum Granules is set to anything other than zero, the PGE is marked as failed because it did not produce the expected number of output granules.
  - If Maximum Granules is set to anything equal to or greater than zero for Output 2, the PGE is scheduled and executes successfully.

The Minimum/Maximum Granules Production Rules are additions to settings/syntax put into the ODL files for other production rules. All Production Rules have a Minimum and Maximum Granule setting for both inputs and outputs, even though both values may be set to one (1).

11.2.4.1 PGE Science Metadata ODL File Parameters

The PGE science metadata ODL file syntax for implementing the Minimum/Maximum Production Rule for **input** data includes the following types of entries:

```
OBJECT = PCF_ENTRY
  .
  PCF_FILE_TYPE =
  .
  .
  MIN_GRANULES_REQUIRED =
  MAX_GRANULES_REQUIRED =
```

.
.
.
END_OBJECT = PCF_ENTRY

Accordingly, the following parameters must be set properly in order to implement the Minimum/Maximum Production Rule:

- PCF_FILE_TYPE.
- MIN_GRANULES_REQUIRED.
- MAX_GRANULES_REQUIRED.

The PCF_FILE_TYPE parameter is defined by integers in the range of 1 to 8 (inclusive). The integers are codes for the following types of files:

- 1 - product input files.
- 2 - product output files.
- 3 - support input files.
- 4 - support output files.
- 5 - user defined runtime parameters.
- 6 - interim/intermediate input files.
- 7 - interim/intermediate output files.
- 8 - temporary input/output.

For inputs (any PCF_ENTRY with a PCF_FILE_TYPE equal to 1, 3 or 6) the following pair of values must be set for each PCF_ENTRY:

- MIN_GRANULES_REQUIRED
  - Minimum number of granules required for the input.
  - A value of zero (MIN_GRANULES_REQUIRED = 0) would mean that the PGE could execute if no granules for that particular input could be found (in effect, the input is an **optional input**).
  - A value of three (for example) would mean that the PGE must have at least three granules of the input before the PGE can be executed.
- MAX_GRANULES_REQUIRED
  - Maximum number of granules for the input that the PGE is able to successfully process.
  - A value of four (for example) would mean that the PGE would process at most four granules for the input.
  - If MAX_GRANULES_REQUIRED = 4 and more than four granules are found for the given input, the PGE is not executed.

The PGE science metadata ODL file syntax for implementing the Minimum/Maximum Production Rule for **output** data includes the following types of entries:

OBJECT = PCF_ENTRY
.
PCF_FILE_TYPE =
.

```
.
MIN_GRANULE_YIELD =
MAX_GRANULE_YIELD =
.
.
.
END_OBJECT = PCF_ENTRY
```

For outputs (any PCF_ENTRY with a PCF_FILE_TYPE equal to 2, 4 or 7) the following pair of values must be set for each PCF_ENTRY.

- MIN_GRANULE_YIELD
  - Minimum number of granules that the PGE produces for the output.
  - A value of zero (MIN_GRANULE_YIELD = 0) means that the PGE produces no granules for the output (the output is an optional output).
  - A value of three (for example) means that the PGE produces at least three granules of the output during a successful execution.
- MAX_GRANULE_YIELD
  - Maximum number of granules that the PGE produces for this output.
  - A value of four (for example) means that at most the PGE produces four granules for the output.
  - Note that sizing of disk space is based on this number, so making it too small could cause problems on the science processor disks.

11.2.5 Optional DPRs Production Rule

The Optional DPRs Production Rule (also called the Data-Scheduled Production Rule) makes the execution of a PGE subject to the availability of a **key input**. The system generates DPRs for every possible instance of the key input data but executes only the DPRs for which data are either produced in data processing or can be acquired from the archive.

The Optional DPRs Production Rule applies to PGEs that process certain kinds of **non-routine data**.

- **Routine Data**
  - Data that can be predicted, that come in at specific intervals and are always of a specified length.
  - Routine data makes it possible for the Basic Temporal Production Rule to schedule PGEs based on their input data.
- **Non-Routine Data**
  - Data that cannot be predicted because they come in at random periods and/or their length is variable.
  - Examples include an "optional" output of an upstream PGE, or data that are archived at random periods (e.g., some forms of ASTER data).

An Optional DPR has as its **key input** a non-routine data type. There are two sets of circumstances that lead to the scheduling of Optional DPRs:

- Every possible time that the input is produced in data processing (i.e., the key input is produced as an "optional" output by an upstream PGE).
- Whenever a new granule (of a particular data type) can be acquired from the archive (e.g., archived data that were inserted at unpredictable times).

NOTE: Key input applies to the Spatial Query Production Rule as well as the Optional DPRs Production Rule. Consequently, it is possible for the Optional DPRs and Spatial Query Production Rules to be used in combination only when they use the same key input.

An example of the first condition starts with a MODIS PGE that produces a certain product only when the input data were collected during the satellite's "Day" mode. A second MODIS PGE is scheduled to use the optional ("Day"-mode) product from the first MODIS PGE as its key input. The second MODIS PGE is scheduled to run after every instance of the first MODIS PGE; however, only the DPRs that can use the optional products resulting from runs of the first MODIS PGE are executed. The remaining DPRs cannot be executed because there is no input data for them.

The second condition is illustrated by ASTER routine processing, which makes use of the Optional DPRs Production Rule to schedule and execute ASTER PGEs for new data that have been archived. (Note that the DAAC ingests and archives ASTER production data from tapes supplied by the ASTER Ground Data System on a frequent but not entirely predictable basis.) When the Production Planner creates a Production Request for an ASTER PGE, the **insertion time** range (i.e., the time period when the desired data were archived) is used as opposed to the **collection time** (when the satellite instrument gathered the data). DPRs specifying the ASTER PGE are scheduled and executed for the data granules that were actually inserted in the archive during the time period specified in the Production Request.

An illustration of the Optional DPRs production rule is presented in Figure 11.2.5-1. In the figure there are two DPRs (i.e., DPR-1 and DPR-2) for the upstream PGE and two DPRs (i.e., OPT-1 and OPT-2) for the PGE subject to the Optional DPRs Production Rule. The "Optional DPRs" PGE takes as input the optional output of the upstream PGE. When it is executed, DPR-1 produces the optional output, so the dependent DPR (OPT-1) is executed. However, OPT-2 is not executed because DPR-2 (on which OPT-2 depends) does not produce the optional output.

The Optional DPRs Production Rule is set up during the SSI&T process. It uses many of the same parameter settings as the Basic Temporal Production Rule so the values specified in the Basic Temporal Production Rule (or other production rules) are set first, then the Optional DPRs Production Rule syntax is added.

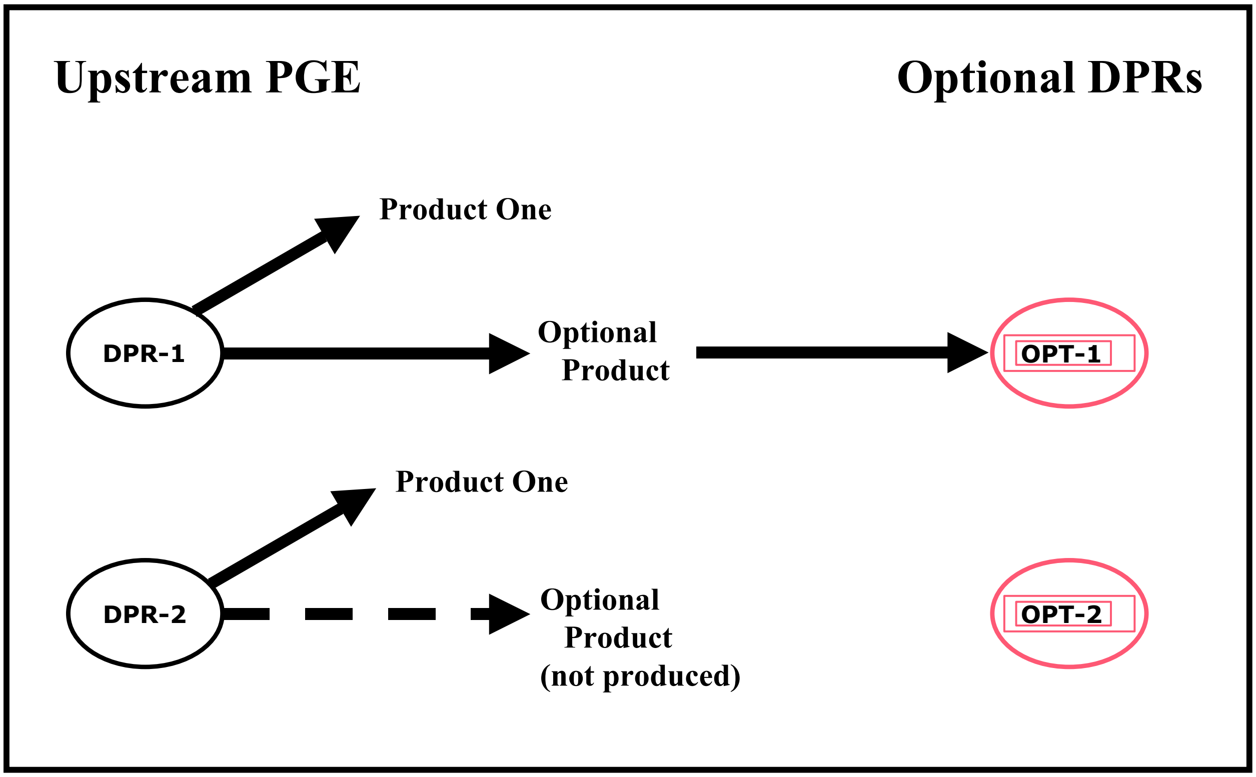

Figure 11.2.5-1. Example of the Optional DPRs Production Rule

11.2.5.1 PGE Science Metadata ODL File Parameters

The following two types of PGE science metadata ODL file entries must be made in order to set up the Optional DPRs Production Rule:

- SCHEDULE_TYPE.
- KEY_INPUT.

The SCHEDULE_TYPE parameter is set as follows:

- SCHEDULE_TYPE = "Data"
  - This demonstrates the appropriateness of the term "Data-Scheduled Production Rule."
  - Other schedule types include Time, Orbit, and Snapshot.

The key input is designated by including the following parameter in the PCF_ENTRY for whichever input is to be the key input:

- KEY_INPUT = "Y"
  - Assigning a value of "Y" to the KEY_INPUT parameter identifies the data as a key input and it is subsequently treated as such.
  - Either assigning a value of "N" to the KEY_INPUT parameter or leaving out the parameter entirely identifies the non-key input data.

- Only one key input is allowed per PGE profile.

The Production Planner's role in the implementation of the Optional DPRs Production Rule was described in the MODIS and ASTER examples previously described and varies with the kind of key input:

- Optional output of an upstream PGE (MODIS example).
  - Production Planner creates Production Requests for the PGE subject to the Optional DPRs Production Rule and specifies the same date/time range as for the upstream PGE.
  - Some of the DPRs generated as a result of the Production Request will never run due to lack of input data.
- Ingested on an irregular time schedule (ASTER example).
  - Production Planner specifies the data insertion time range when creating Production Requests.
  - All DPRs generated as a result of the Production Requests should be capable of running.

11.2.6 Metadata Checks and Metadata Query Production Rules

The Metadata Checks and Metadata Query Production Rules are similar in definition and use. Both production rules allow the PGE to specify granule-level metadata values that define whether the PGE can accept one (or more) of its inputs. The rules differ only in the results of metadata search performed.

- Metadata Checks Production Rule.
  - When PLS requests the Science Data Server to search for the input(s), the Science Data Server "checks" the metadata of all granules that match the time frame with respect to the value(s) allowed by the PGE.
  - If any granule fails to match the specified value(s), the PGE is not executed.
- Metadata Query Production Rule.
  - When PLS requests the Science Data Server to search for the input(s), the Science Data Server adds to the query the metadata value(s) desired by the PGE.
  - Only the granules that match the time frame of the PGE plus the granule-level metadata value(s) specified by the PGE are staged for the PGE to use as input.
  - If no granules are found matching the conditions and the input is not optional, the PGE is not executed.
- Example of Metadata Checks:
  - A MODIS PGE is run when the Percent Cloud Cover of its inputs is greater than 25 percent.
  - The Metadata Checks Production Rule is used to specify the granule-level metadata value of greater than 25.
  - When the PGE is scheduled and is ready to start, two granules match the timeframe of the Production Request for the input with the Metadata Check.
  - If both granules have a Percent Cloud Cover greater than 25 percent, execution of the PGE starts and both granules are staged.

- If one of the granules has a Percent Cloud Cover of 15 percent, the PGE is not executed.

- Example of Metadata Query:
  - A MODIS PGE is run when as many granules as possible of one of its inputs have a QA Value = "Good".
  - The Metadata Query Production Rule is used to specify the granule-level metadata value = "Good".
  - When the PGE is scheduled and is ready to start, two granules match the time frame of the production request for the input with the Metadata Query.
  - If both granules have a QA Value = "Good", execution of the PGE starts and both granules are staged.
  - If one of the granules has a QA Value = "Bad", the PGE executes but with only one granule (the one with QA Value = "Good").

The Metadata Checks and Metadata Query Production Rules are used in conjunction with the times specified in the Basic Temporal Production Rule or other production rules. The Metadata Check or Query is added information that further refines what granules are sought by the PGE.

The Metadata Checks and Metadata Query Production Rules work differently depending on whether the Production Request is created for a time in the past or for a time in the future.

- Past data.
  - The Production Request Editor performs the Metadata Query or Metadata Check immediately.
  - Depending on what other production rules may be applicable, if no data is found to match the “query” (or data is found that does **not** match the “check”), the DPR fails to be created.
- Future data.
  - The Metadata Query/Metadata Check is put off until the DPR Data Collection Stop Time passes **plus** the value defined in the ODL for QUERY_DELAY.
  - The delay allows the Metadata Query/Metadata Check to be put off until it's likely that all matching data will be present.

Multi-Granule ESDTs are a special case of the Metadata Query Production Rule. Multi-Granule ESDTs are used for PGE inputs or outputs when more than one granule of the same ESDT exists for the same temporal range (time period). The Multi-Granule ESDT mechanism employs a metadata parameter to differentiate between the “equal in time” granules. A metadata parameter is selected that is unique across granules for the same time period and that is used by PDPS to keep track of which granule is which when the granules are produced. Later, if only one pair of granules for a particular time period is needed as input to the PGE, the Metadata Query is used to ensure that PDPS schedules the correct granule as input.

Data Day is actually an addition to the Metadata Query Production Rule involving **runtime parameter** values. Two settings (i.e., “Start Data Day” and “End Data Day”) allow a PGE to perform a Metadata Query for the start of the Data Day and the end of the Data Day. The Start Data Day and End Data Day are calculated by subtracting twelve (12) hours from the starting day for which the PGE is executing and adding twelve (12) hours onto the ending day for which the PGE is running. So if the PGE is running from 00:00:00 on 07/04 to 00:00:00 07/05 then the Start Data Day = 07/03 12:00:00 and the End Data Day = 07/06 12:00:00.

Using runtime parameter values is a capability of the Metadata Query and Metadata Checks Production Rules. Rather than use a hard-coded value for the check or query, a value computed from one of the other production rules can be used.

Figure 11.2.6-1 illustrates the Metadata Checks and Metadata Query Production Rules. If no Metadata Check or Query were applicable, the PGE shown in the figure would use three granules of input (i.e., Granules A through C). However, let us assume that the metadata value to be checked/queried is %CloudCover. Each granule has a different value for %CloudCover.

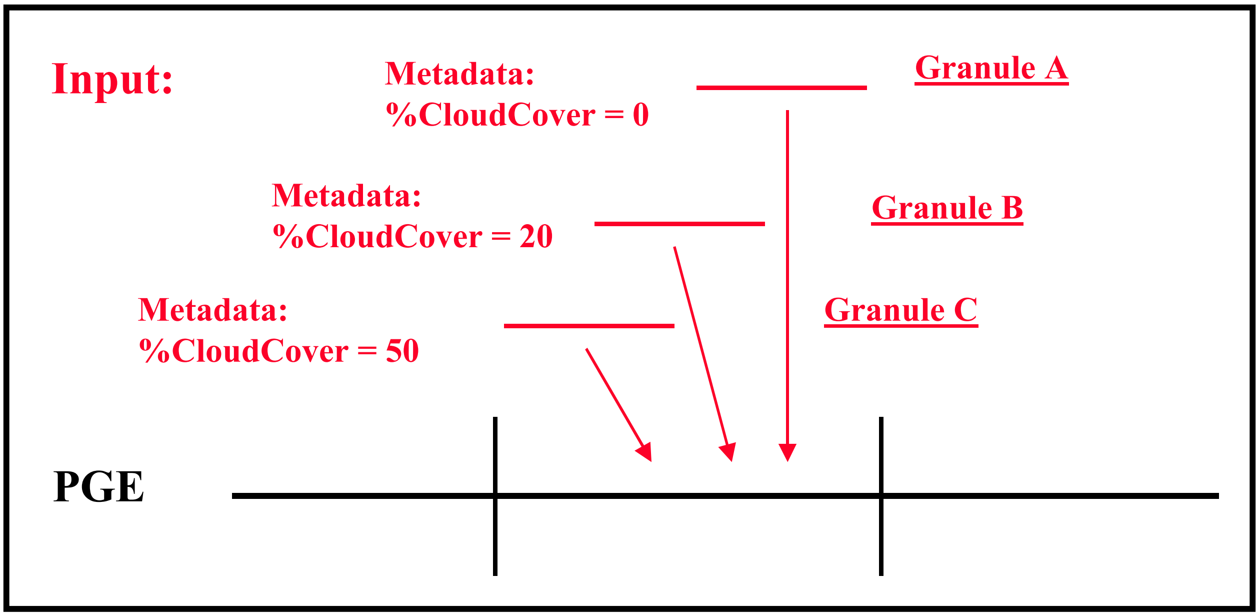

Figure 11.2.6-1. Example of the Metadata Checks and Query Production Rules

The following results demonstrate the differences between the Metadata Checks and Metadata Query Production Rules, especially with respect to the number of inputs that the PGE receives when different values are specified:

- Metadata Check of %CloudCover < 80:
  - In this case all three granules are acquired and the PGE is scheduled and executed.
- Metadata Query of %CloudCover < 80:
  - All three granules are acquired and the PGE is scheduled and executed.
- Metadata Check of %CloudCover = 50:
  - The PGE is not scheduled because only one of the three granules (Granule C) meets the criterion.

- Metadata Query of %CloudCover = 50:
  - Granule C is found and if the PGE's Min/Max Granules parameters are set to allow one granule, that one granule is acquired and the PGE is scheduled and executed.
- Metadata Check of %CloudCover < 50:
  - The PGE is not scheduled because only two of the three granules (Granule A and B) meet the criterion.
- Metadata Query of %CloudCover < 50:
  - Granules A and B are found and if the PGE's Min/Max Granules parameters are set to allow two granules, the granules are acquired and the PGE is scheduled and executed.
- Metadata Check of %CloudCover <= 50:
  - The PGE is scheduled and executed because all three granules meet the criterion.
- Metadata Query of %CloudCover <= 50:
  - All three granules are found and acquired and the PGE is scheduled and executed.
- Metadata Check of %CloudCover = 20:
  - The PGE is not scheduled because only one of the three granules (Granule B) meets the criterion.
- Metadata Query of %CloudCover = 20:
  - Granule B is found and if the PGE's Min/Max Granules parameters are set to allow one granule, the granule is acquired and the PGE is scheduled and executed.
- Metadata Check of %CloudCover < 20:
  - The PGE is not scheduled because only one of the three granules (Granule A) meets the criterion.
- Metadata Query of %CloudCover < 20:
  - Granule A is found and if the PGE's Min/Max Granules parameters are set to allow one granule, the granule is acquired and the PGE is scheduled and executed.
- Metadata Check of %CloudCover = 10:
  - The PGE is not scheduled because none of the three granules meets the criterion.
- Metadata Query of %CloudCover = 10:
  - The PGE is not scheduled because no granules are returned from the query (unless Minimum Granules is set to 0).

Note that there can be more than one Metadata Check or Metadata Query on a given input. In the preceding example, a Metadata Check on %CloudCover can be combined with a Metadata Query on another parameter to further limit the input.

The Metadata Checks and Metadata Query Production Rules are additions to settings/syntax put into the ODL files for other production rules. The addition of a Metadata Check or a Metadata Query to an input means that other production rules used to evaluate that input will be applied in combination with the Metadata Check or Metadata Query.

11.2.6.1 PGE Science Metadata ODL File Parameters

At the main ODL level (not inside any objects) both the Metadata Checks and Metadata Query Production Rules use the following parameter to determine when to perform the check/query:

- QUERY_DELAY

The parameter is a positive integer (number of seconds) that defines the amount of time to wait before performing the Metadata Query or Metadata Check. When planning DPRs to run on future data, the Production Request Editor sets a timer for all Metadata Checks/Queries at the Data Collection Stop Time (end of the DPR) **plus** the value specified for the QUERY_DELAY parameter. The value adds time to the Query/Check so the PGE can be assured that all matching data will be present when the Query/Check is performed.

Both production rules use also need the following setting in the PCF_ENTRY to ensure that Planning and Data Processing consider metadata values when acquiring input data:

```
OBJECT = PCF_ENTRY
  .
  .
  QUERY_TYPE = "Metadata"
  .
END_OBJECT = PCF_ENTRY
```

QUERY_TYPE tells Planning and Data Processing how to retrieve the data for the input. For inputs based on metadata (whether Metadata Check or Metadata Query), the value is set to "Metadata".

Although the Metadata Checks and Metadata Query Production Rules are similar, there are two different ODL objects used to define them within a PCF_ENTRY in the PGE science metadata ODL file; i.e., the METADATA_CHECKS object and the METADATA_QUERY object.

The METADATA_CHECKS object has the following syntax:

```
OBJECT = PCF_ENTRY
  .
  .
  .
  .
  .
  OBJECT = METADATA_CHECKS
    .
    .
    .
  END_OBJECT = METADATA_CHECKS
END_OBJECT = PCF_ENTRY
```

The METADATA_QUERY object has the same syntax except “METADATA_QUERY” replaces “METADATA_CHECKS” in every instance.

Most of the following parameters must be set in the PGE science metadata ODL file within the METADATA_CHECKS or METADATA_QUERY ODL object (as applicable) in order to implement either the Metadata Checks or Metadata Query Production Rule:

- CLASS.
- PARM_NAME.
- OPERATOR.
- VALUE.
- DATABASE_QUERY.
- KEY_PARAMETER_NAME (optional).
- KEY_PARAMETER_VALUE (optional).

CLASS is a simple counter used to differentiate the different Metadata Checks or Metadata Query objects within the file. Since each Metadata Checks or Metadata Query object resides within a different PCF_ENTRY object, the CLASS for an METADATA_CHECKS or METADATA_QUERY object can always be 1 (e.g., CLASS = 1).

PARM_NAME is the name of the metadata parameter on which the check or query is to be performed. The value specified for PARM_NAME (e.g., PARM_NAME = "%CloudCover") must be part of the granule-level metadata of the ESDT. In addition, it must match the parameter name specified in the ESDT science metadata ODL file.

OPERATOR is the operator (e.g., OPERATOR = "==") on which the check/query is to be performed. The following values are valid for OPERATOR:

- ">"
  - Value in metadata must be greater than.
- "<"
  - Value in metadata must be less than.
- ">="
  - Value in metadata must be greater than or equal to.
- "<="
  - Value in metadata must be less than or equal to.
- "=="
  - Value in metadata must be equal to.
- "!="
  - Value in metadata must be **not** equal to.

VALUE is the value (e.g., VALUE = 50) against which the metadata parameter (defined by PARM_NAME) is compared (using the operator specified by the OPERATOR parameter). The value for the VALUE parameter should be the type of data (e.g., integer or string) as defined in the ESDT ODL metadata for the parameter.

DATABASE_QUERY indicates whether the value for the Metadata Check or Query should be retrieved from the PDPS database rather than through the use of the VALUE parameter. Specifying DATABASE_QUERY permits **runtime parameter values** to be used for Metadata Query or Metadata Checks. The following values are valid for the DATABASE_QUERY parameter:

- "NONE"
  - Use the value in the VALUE parameter; no value from the PDPS database is used.
- "PATH NUMBER"
  - Use the Path Number (0-233) of the orbit for which the PGE is scheduled.
- "ORBIT NUMBER"
  - Use the Orbit Number of the orbit for which the PGE is scheduled.
- "TILE ID"
  - Use the Tile ID of the current Data Processing Request.

KEY_PARAMETER_NAME is an optional parameter that is used to specify the container within a multi-container metadata group (i.e., the MeasuredParameters metadata group in most ESDTs). The KEY_PARAMETER_NAME (e.g., KEY_PARAMETER_NAME = "ParameterName" for metadata checks or queries within the MeasuredParameters group) in conjunction with the KEY_PARAMETER_VALUE allows PDPS to determine which container within the multi-container group is to be the object of the check or query. KEY_PARAMETER_NAME is **not** used for product-specific attributes.

KEY_PARAMETER_VALUE is an optional parameter that is used to specify the **value** (e.g., KEY_PARAMETER_VALUE = "LandCoverage") for the container within a multi-container metadata group (i.e. the MeasuredParameters metadata group in most ESDTs). The

KEY_PARAMETER_VALUE in both the PGE science metadata ODL file and ESDT science metadata ODL file must match.

For Multi-Granule ESDT inputs the METADATA_QUERY object (consisting of entries for CLASS, PARM_NAME, OPERATOR, VALUE, DATABASE_QUERY, KEY_PARAMETER_NAME, and KEY_PARAMETER_VALUE) is replaced by the following parameter under the PCF_ENTRY object in the PGE science metadata ODL file:

- DISTINCT_VALUE.

The DISTINCT_VALUE must be set to the value of the metadata parameter that is used to differentiate granules within the Multi-Granule ESDT. In addition, the input or output defined by the PCF_ENTRY must have a corresponding DISTINCT_PARAMETER entry in the appropriate ESDT science metadata ODL file.

11.2.6.2 ESDT Science Metadata ODL File Parameters

The METADATA_DEFINITION ODL object surrounds the definition for Metadata Checks or Metadata Query information within the ESDT science metadata ODL file. An OBJECT/END_OBJECT pair is needed to separate the parameters defining the Metadata Definition from the rest of the parameters defining the ESDT with the following syntax:

```
OBJECT = METADATA_DEFINITION
 .
 .
```

.

END_OBJECT = METADATA_DEFINITION

A METADATA_DEFINITION object can match multiple Metadata Checks or Metadata Query objects in various PGE science metadata ODL files. There is no difference between the two production rules with respect to the parameters that need to be set in the ESDT science metadata ODL file. Most of the following parameters must be set:

- CLASS.
- PARM_NAME.
- CONTAINER_NAME.
- TYPE.
- KEY_PARAMETER_NAME (optional).
- KEY_PARAMETER_VALUE (optional).

CLASS is a simple counter used to differentiate the different Metadata Definition objects within the file. Each Metadata Definition object within the file must have a **different** CLASS value.

PARM_NAME is the name of the Metadata parameter on which the check or query will be performed. The value specified for PARM_NAME must be part of the granule-level metadata of the ESDT. It must also match the parameter name specified in the PGE science metadata ODL file(s).

CONTAINER_NAME is the name of the Metadata Group within which the metadata parameter defined by PARM_NAME is contained. For product-specific attributes CONTAINER_NAME is set to the string "AdditionalAttibutes" (i.e., CONTAINER_NAME = "AdditionalAttibutes").

TYPE indicates the type of data within the metadata parameter. The following values are valid for TYPE:

- "INT"
  - Integer data.
- "FLOAT"
  - Floating point data.
- "STR"
  - String or character data.
  - Note that dates and times are considered string data.

KEY_PARAMETER_NAME and KEY_PARAMETER_VALUE are optional parameters that are used to specify the value for the container within a multi-container metadata group (i.e., the MeasuredParameters metadata group in most ESDTs). They allow PDPS to determine which container within the multi-container group is to be the object of the check or query. There must be a match between the KEY_PARAMETER_VALUE in both the ESDT science metadata ODL file and the PGE science metadata ODL file for any PCF_ENTRY that uses the ESDT.

The ESDT science metadata ODL file for an input specifying Multi-Granule ESDTs needs to have the following parameter added:

- DISTINCT_PARAMETER.

The DISTINCT_PARAMETER must be set to the name of the metadata parameter that is used to differentiate granules within the Multi-Granule ESDT. A corresponding METADATA_DEFINITION must be created to help PDPS find the specified metadata parameter when querying the Science Data Server.

11.2.7 Spatial Query Production Rule

The Spatial Query Production Rule allows a PGE to select input(s) based on the spatial coverage of another input (called the **key input**). The PDPS queries the Science Data Server for the spatial coverage of the key input, then uses it in acquiring any subsequent inputs that the PGE has requested that have the same spatial coverage.

- Example:
  - Level 0 input data for an ASTER DPR covers a small section of the Earth.
  - The PGE requires ancillary data that covers the same area to complete its processing.
  - The PGE uses the Spatial Query Production Rule to mark the geographic input as its key input.
  - The PGE specifies that the ancillary input is to be retrieved for the same spatial coverage as that of the key input.
  - When PDPS finds an input granule for the PGE, it performs a Spatial Query to acquire the ancillary input with the same spatial coverage as that of the key input.

Without specifying coordinates, PDPS can match inputs against the spatial constraint of the key input, and give to a PGE only those granules, which overlap in area.

Spatial Pad is an addition to the Spatial Query Production Rule. Spatial Pad is a means of padding the spatial constraints of the key input. The specified pad is added to all sides of the key input's spatial shape. All granules that intersect the expanded area are retrieved.

Due to the limitations in the current version of the SQS COTS package, the spatial region defined (i.e., the original size of the polygon combined with the size of the pad) must not exceed 60 degrees in latitude and longitude. If no spatial pad is specified or the pad is equal to zero, normal spatial query is used.

The **Spatial Query** Production Rule (unlike most time-base production rules) will work with both static and dynamic granules. Thus the Digital Elevation Models (DEMs) can be retrieved via a Spatial Query.

Figure 11.2.7-1 is an illustration of the Spatial Query Production Rule. The figure shows a PGE that has two input types, one of which is the key input. The other type of input has granules labeled with the names of various colors. One granule (i.e., "green") of the key input is found. The spatial coordinates of the granule are retrieved and all inputs of the second ESDT are checked for overlap with the key input's coordinates.

Assuming that all granules relate to the same time period, the granules are evaluated as follows:

- The "yellow" granule is not retrieved as an input because its spatial coordinates do not overlap with those of the key input.

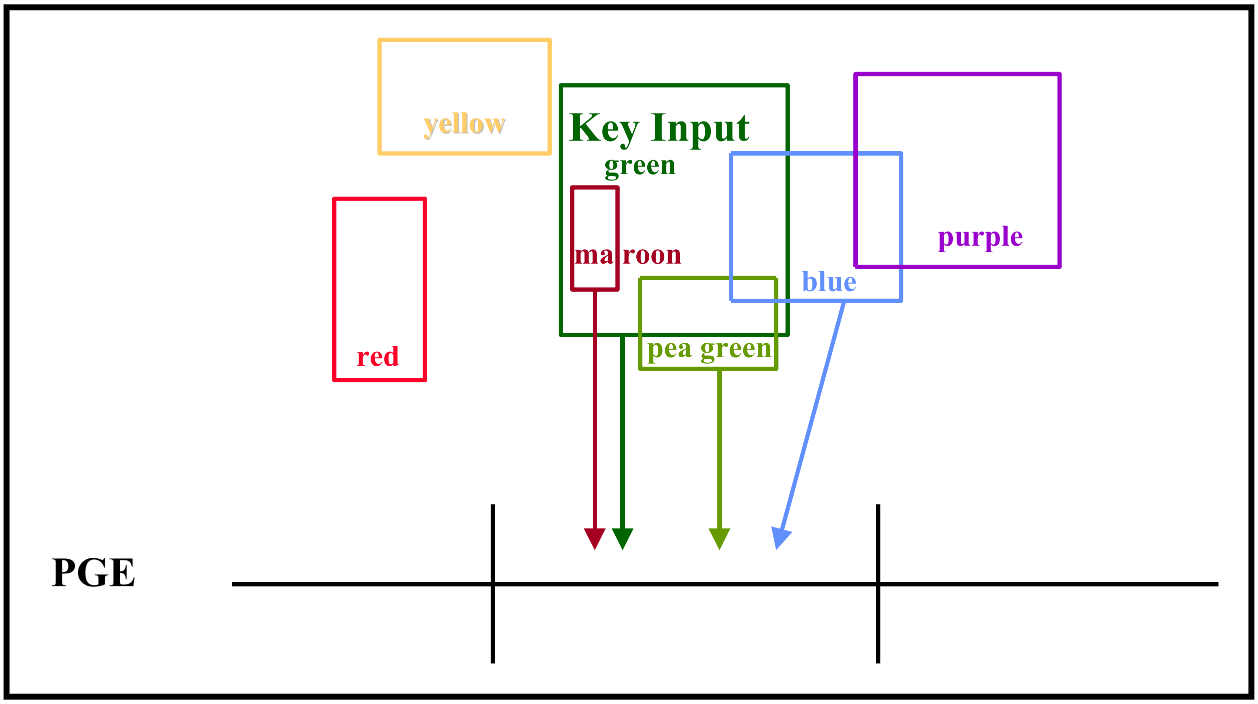

Figure 11.2.7-1. Example of the Spatial Query Production Rule

- The “red” granule is not retrieved as an input because its spatial coordinates do not overlap with those of the key input.
- The “blue” granule is retrieved as an input because its spatial coordinates overlap with those of the key input. Part of its spatial constraint is within the constraint of the key input.
- The “maroon” granule is retrieved as an input because its spatial coordinates overlap with those of the key input. The spatial constraint of this granule is completely within the constraint of the key input.
- The “pea green” granule is retrieved as an input because its spatial coordinates overlap with those of the key input. Part of its spatial constraint overlaps with that of the key input.
- The “purple” granule is not retrieved as an input because its spatial coordinates do not overlap with those of the key input. It does not matter that it overlaps with another input that is accepted (i.e., the “blue” granule).

The Spatial Query Production rule is somewhat of an addition to other production rules. As such, it needs the same parameter settings as the Basic Temporal Production Rule. The values specified in the Basic Temporal Production Rule (or other production rules) are set first, then the Spatial Query Production Rule syntax is added.

11.2.7.1 Spatial Query with Spatial Pad

Figure 11.2.7-2 is an illustration of Spatial Query with Spatial Pad. As in Figure 11.2.7-1 the PGE has two input types, one of which is the key input (in green). The Spatial Pad value (the dotted line in Figure 11.2.7-2) is added around the key input to increase the area covered by the spatial coordinates. Then all inputs of the second ESDT are checked for overlap with the padded coordinates.

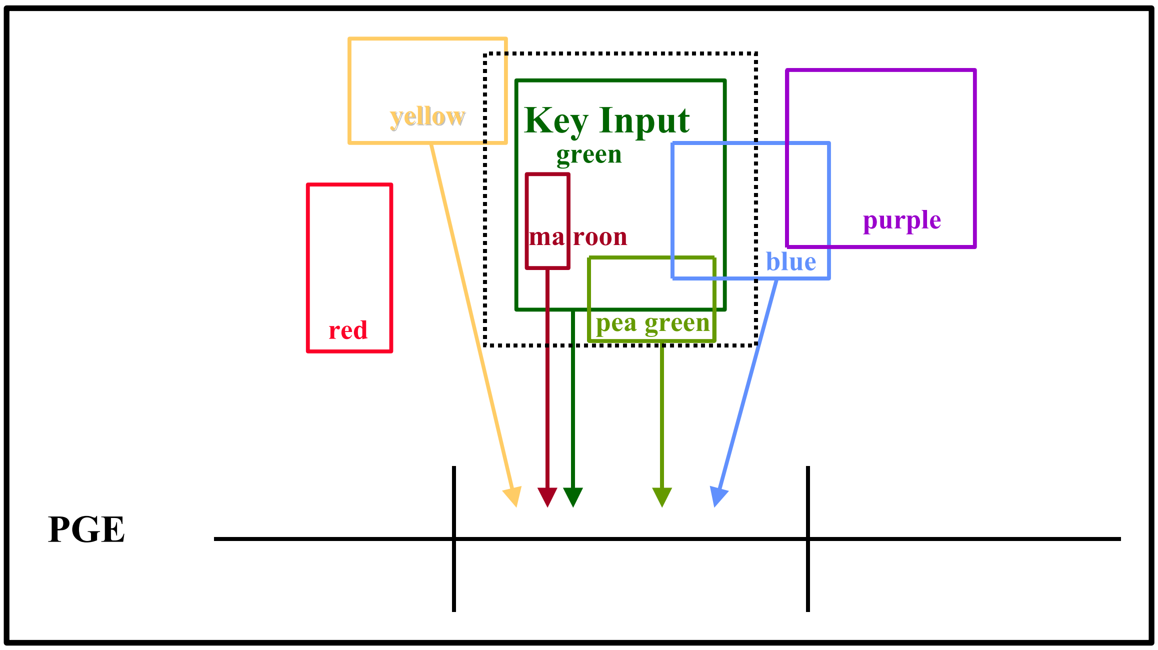

Figure 11.2.7-2. Example of the Spatial Query Production Rule with Spatial Pad

Assuming that all granules relate to the same time period, the granules are evaluated as follows:

- The "yellow" granule is retrieved as an input; although its spatial coordinates do not overlap with those of the key input, they do overlap the padded area (the dotted line).
- The "red" granule is not retrieved as an input because its spatial coordinates do not overlap with either the key input or the pad.
- The "blue" granule is retrieved as an input because its spatial coordinates overlap with the key input. Part of its spatial constraint is within the constraint of the key input.
- The "maroon" granule is retrieved as an input because its spatial coordinates overlap with the key input. The spatial constraint of this granule is completely within the constraint of the key input.

- The “pea green” granule is retrieved as an input because its spatial coordinates overlap with the key input. Part of its spatial constraint overlaps with that of the key input.
- The “purple” granule is not retrieved as an input because its spatial coordinates do not overlap with the key input. It does not matter that it overlaps with another input that is accepted, the “blue” granule.

11.2.7.2 PGE Science Metadata ODL File Parameters

Each input file is defined by a separate PCF_ENTRY object in the PGE science metadata ODL file. To mark a particular input as being subject to a Spatial Query the QUERY_TYPE parameter needs to be set to “Spatial” within that input’s PCF_ENTRY:

```
OBJECT = PCF_ENTRY
   .
   .
   .
   QUERY_TYPE = "Spatial"
   .
   .
   .
END_OBJECT = PCF_ENTRY
```

QUERY_TYPE indicates what type of query is to be done to acquire the input defined by the PCF_ENTRY object. Valid values are as follows:

- "Temporal" - Input is acquired based on time.
  - The Basic Temporal and/or the Advanced Temporal Production Rules is/are used to get the input.
  - “Temporal” is the value that is assumed if the parameter is left out of the PCF_ENTRY object.
- "Spatial" - Input is acquired based on spatial coordinates (as well as time).
  - An input must be designated the key input to be used in determining the spatial constraints of the search.
  - “Spatial” is the value specified for each input that uses the Spatial Query Production Rule.
- "Tile" - Input is acquired by the spatial definition of a tile.
  - Refer to the Tiling Production Rule for additional information.
- "Already Created Tile" - Input is acquired based on the tile ID of an already created tile.
  - Refer to the Tiling Production Rule for additional information.

The PCF_ENTRY object for the key input includes the following syntax:

```
OBJECT = PCF_ENTRY
   .
   .
```

```
    KEY_INPUT = 'Y'
    .
    .
END_OBJECT = PCF_ENTRY
```

The KEY_INPUT is the input on which the spatial queries for other inputs will be based. When a KEY_INPUT parameter is assigned a value of "Y" the corresponding input is designated a key input and is treated as such. A value of "N" or leaving out the parameter entirely specifies a non-key input. Only one (1) key input is allowed per PGE Profile.

If a pad is designated as an addition to the spatial constraints of the KEY_INPUT, the SPATIAL_PAD parameter must be set. The pad value is added to the key input's spatial constraints. The value for the SPATIAL_PAD parameter is set within the PCF_ENTRY defining the key input using the following syntax:

```
OBJECT = PCF_ENTRY
    .
    .
    KEY_INPUT = 'Y'
    SPATIAL_PAD =
    .
    .
END_OBJECT = PCF_ENTRY
```

The SPATIAL_PAD represents the number of kilometers (from 0.0 to 1000.0) that should be added to the Spatial Query. The value is a real number between 0 and 1000 and if not specified, it is set to 0.0 (and thus no pad is added to the spatial queries).

To allow for a temporal offset when performing a spatial query, the SPATIAL_TIME_DELTA parameter is specified to add or subtract time (in seconds) from the query. Consequently, the temporal range for the query may be different from the temporal range of the key input. The value for the SPATIAL_TIME_DELTA parameter is set within the PCF_ENTRY for the spatial input:

```
OBJECT = PCF_ENTRY
    .
    .
    .
    QUERY_TYPE = "Spatial"
    SPATIAL_TIME_DELTA =
    .
    .
END_OBJECT = PCF_ENTRY
```

The SPATIAL_TIME_DELTA adds or subtracts seconds to/from the start and stop times of the DPR when performing the spatial query for the input defined by the PCF_ENTRY. In effect it widens the temporal query. For example…

- DPR Time = 2:00:00 to 3:00:00
- SPATIAL_TIME_DELTA = 20 [seconds]
- Query time for spatial input: 1:50:40 to 3:00:20

11.2.8 Closest Granule Production Rule

The Closest Granule Production Rule allows a PGE to request the nearest input granule from the Data Processing Request time. The PDPS requests a search forward or backward for a specified period of time until it finds a granule that matches the request. However, there is a limit to the number of queries that are performed. The number of queries and the period length of the query are specified during SSI&T.

- Example:
  - A PGE processes data at daily intervals and could use a particular type of calibration granule that would allow it to determine the nearest parameters of the instrument.
  - Although most calibration coefficients are defined as static granules, in this case there is a dynamic granule that is received about once a month.
  - The closest such granule would be optimal, so the PGE uses the Closest Granule Production Rule to search forward or backward from the time of the DPR to find the nearest calibration granule.

The Closest Granule Production Rule supersedes the Most Recent Granule Production Rule. The latter allowed the search for inputs to go backward in time from the start of the DPR. The Closest Granule Production Rule allows the search for input granules to go either backward or forward in time, increasing the flexibility of the rule. The Closest Granule Production Rule has all of the ability of Most Recent Granule, plus the ability to search forward in time for input data.

The Closest Granule Production Rule uses three values to determine what to query. The three values are concerned with the period of time to query, the direction of the query, and the number of queries allowed.

- Period (Offset).
  - Tells the PDPS software the query duration.
- Direction.
  - Indicates whether the query goes forward (positive) or backward (negative) in time.
  - In the PlDataTypeReq table in the PDPS database, the Direction and the Period information are combined, so the sign of value (+ or -) indicates the Direction and the magnitude indicates the Period.
- Maximum Number of Queries.
  - Tells the PDPS software how many time periods (as defined by the Offset) to search (either forward or backward in time) for a matching granule.

The PDPS does a Basic Temporal query before using Closest Granule to find the input. If the desired input is not found within the time period of the DPR, PDPS performs a query (in the specified direction) against the Science Data Server for the period defined by the offset. Again, if no matching granule is found, PDPS repeats the query, going backward or forward in time by the value specified in the offset. If no acceptable granule has been found before the maximum number of queries is reached, PDPS fails to generate the DPR due to insufficient input data. However, there is a special case for the forward search; i.e., when the next search interval exceeds the current time, the search stops at the current time.

If the DPR is planned for a future time, the DPR is created using place-holder granules and a timer is activated. When the current time reaches the stop time of DPR, the timer invokes the Closest Granule method to search for the actual granules, which are then used to replace the dummy granules.

Figure 11.2.8-1 illustrates the Closest Granule Production Rule. In the example, the PGE has a boundary of “start of day” and a period of one hour, so it is scheduled to run for one hour’s worth of input data. The input has a period of one hour, and can come in at any hour of the day. Consequently, the PGE requests one granule of input.

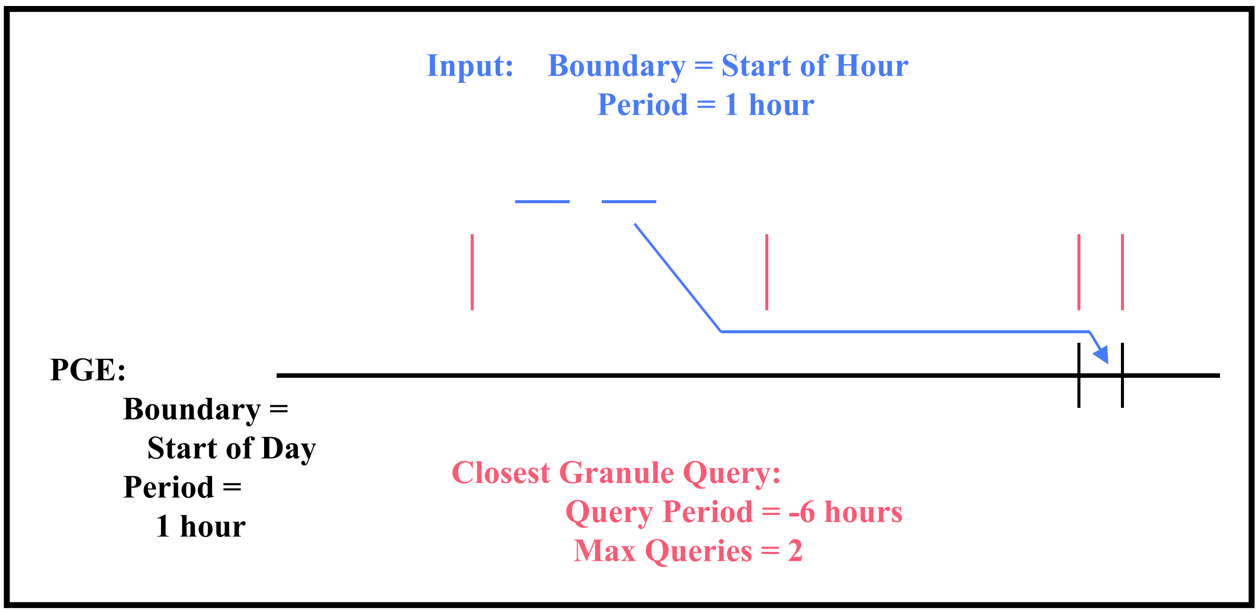

Figure 11.2.8-1. Example of Closest Granule Production Rule

The PGE has defined the Closest Granule Production rule with a 6-hour period and a direction of backward, meaning that it queries back in time in six-hour intervals. The maximum number of queries is two. The PDPS performs a query for the input based on the time period of the DPR. Not finding any matching data, it uses the Closest Granule information to query for a six-hour period beginning six hours before the start time of the DPR. Again nothing is found, so a second Closest Granule query is performed, this one six hours before the last Closest Granule query. The second query results in the discovery of two matching granules. The PDPS selects the granule that is later in time and schedules the PGE to use it as input.

If the Closest Granule Production Rule were used in conjunction with the Minimum/Maximum Number of Granules Production Rule, it might be possible for both granules to be selected in the previously described Closest Granule query. If the example included setting the Maximum Number of Granules to two, both granules would be selected as input to the PGE.

The Closest Granule Production Rule needs the same parameter settings as the Basic Temporal Production Rule. The values needed for the Basic Temporal Production Rule must be set before the Closest Granule Production Rule syntax is added.

11.2.8.1 PGE Science Metadata ODL File Parameters

In addition to the parameter settings for the Basic Temporal Production Rule, the following parameters must be set within the appropriate PCF_ENTRY in the PGE science metadata ODL file in order to implement the Closest Granule Production Rule:

- CLOSEST_QUERY_OFFSET.
- CLOSEST_QUERY_RETRIES.
- CLOSEST_QUERY_DIRECTION.

As previously mentioned, if it is necessary to distinguish among granules with the same ESDT (e.g., among granules associated with different sensors for an instrument), the following parameter must be added to the PCF_ENTRY in the PGE science metadata ODL file:

- DISTINCT_VALUE.

CLOSEST_QUERY_OFFSET is the offset added to or subtracted from the Data Start Time of the DPR and uses as the query for the requested input data type. The specified value has the format "<Period Type>)=<Length of Period>" (e.g., CLOSEST_QUERY_OFFSET = “HOURS=6”). The following “Period Type” values are used in implementing the Closest Granule rule:

- "WEEKS"
  - Offset is some number of weeks.
  - For example, "WEEKS=2" would be a 14-day offset.
- "DAYS"
  - Offset is some number of days.
  - For example, "DAYS=5" would be a 120-hour offset.
- "HOURS"
  - Offset is some number of hours.
  - For example, "HOURS=4" would be a 240-minute offset.
- "MINS"
  - Offset is some number of minutes.
  - For example, "MINS=5" would be a 300-second offset.
- "SECS"
  - Offset is some number of seconds.

CLOSEST_QUERY_RETRIES is the maximum number of Closest Granule queries before the DPR fails due to insufficient input data. The specified value is an integer value that is limited only by the maximum size of an integer on the executing hardware.

CLOSEST_QUERY_DIRECTION is the direction ("Forward" or "Backward") in time to search the CLOSEST_QUERY_OFFSET for a matching granule. The following “Direction” values are used in implementing the Closest Granule rule:

- "Forward"
  - Search forward in time (from the end of the DPR).
- "Backward"
  - Search backward in time (from the start of the DPR).

Note that the longer the offset value or the greater the number of retries, the more time that each query requires due to search time at the Science Data Server and processing time of any granules returned. The combination of a large offset with a large number of retries can (if no data granules are found) consume a lot of time while failing to generate a DPR.

The DISTINCT_VALUE (if used) must be set to the value of the metadata parameter that is used to differentiate granules within the Multi-Granule ESDT. In addition, the input or output defined by the PCF entry must have a corresponding entry identifying the metadata parameter in the ESDT science metadata ODL file.

11.2.9 Orbital Processing Production Rule

The Orbital Processing Production Rule is similar to the Basic Temporal Production Rule in that both define the time period for the inputs and outputs of the PGE. The difference is that the Orbital Processing Production Rule uses the orbit of the spacecraft to determine that time period. A PGE that processes data related to every orbit of a satellite uses data related to a time period that is computed from the orbit of that satellite.

- Example:
  - A PGE processes Level 0 data related to each orbit of the Terra satellite.
  - The Terra satellite has an orbital period of 98 minutes so the PGE is scheduled to process data for each 98-minute interval.
  - Since Level 0 data are received every two hours, the data staged for the PGE include every Level 0 granule that falls within the 98-minute PGE interval.
  - Only one granule of Level 0 data is relevant to some 98-minute orbits.
  - Two granules of Level 0 data are relevant to other 98-minute orbits.

The Orbital Processing Production Rule uses the “period” and “boundary” concept just like the Basic Temporal Production Rule. The difference is that for Orbital Processing, the orbit of the spacecraft is taken into account when a PGE or its data are marked as **orbit scheduled**.

When responding to a Production Request for orbit-scheduled processing, PDPS determines the orbit of the satellite via information provided during the SSI&T process. The information (stored in the PDPS database) gives the start time and length of a particular orbit or set of orbits. PDPS extrapolates (or interpolates in the case of an orbit between two orbital periods stored in the database) the start and end times of the PGE that is specified in the Production Request. Data are sought on the basis of the derived start and stop times and the appropriate data granule(s) is/are staged before the PGE is executed.

Orbit model is a model of the satellite's orbits that allows PDPS to perform extrapolations for the Orbital Processing Production Rule. The model is a combination of a database table and a simple algorithm. The database table stores Orbit Number/Orbit Start Time/Orbital Period combinations. The algorithm uses the data to compute the same type of data relevant to subsequent orbits. For example, if Orbit Number 1000 is defined in the database table, the algorithm can calculate the start and end times for Orbits 1001, 1015, 2000, etc.

The Orbit Model works by extrapolation. It is unable to calculate data for any orbit that precedes the earliest entry in the database table. Data for the Orbit Model is specified in the Orbit Model ODL file, which is read only if the PGE requires orbital information.

Orbital path is the path of the satellite over the Earth. It is a number from 0-233 that indicates the region of the Earth covered by a particular orbit. Note that because of the implementation of Orbital Path, there needs to be a mapping between the orbital path calculated by PDPS and the orbital path number expected by the PGEs.

Runtime parameters can be set to values associated with Orbital Processing. The following list of orbital parameters can be placed under runtime parameters:

- Orbit Number.
  - The number of the orbit (starting from zero) and continually increasing.
- Orbital Path Number.
  - The number of the path that maps to the orbit number.
  - The orbital path number is the 0-233 orbital path traversed by the satellite.
- Orbit Number within the Day.
  - The number of the orbit within the given day.
  - It includes any orbit that ends within the given day.
- Granule Number within the Orbit.
  - The number of the granule within a given orbit.
  - It includes any granule that starts within the given orbit.

Figure 11.2.9-1 provides an illustration of the Orbital Processing Production Rule. The PGE in the diagram takes a two-hour input, but is scheduled based on the orbit time and period of the satellite. PDPS uses the data collected at SSI&T to predict the time of the orbit and performs the query to the Science Data Server for the input based on that extrapolated or interpolated orbital time. Granules of input data are allocated to DPRs based on their ability to cover the time period relevant to the DPR.

In the example shown in Figure 11.2.9-1 the length of an orbit is less than the period of the two-hour input, so sometimes a single granule may cover the input time range of a PGE execution and at other times two granules are required. The production rule would work equally well if the data were of a shorter period (e.g., 1/2 hour) than the orbit of a satellite (e.g., 90 minutes). In such a case three granules would be staged for every execution of the PGE.

The Orbital Processing Production Rule is based (at least for the PGE science metadata ODL file) on the same fields used for the Basic Temporal Production Rule. However, the values specified for the parameters provide orbit information rather than time-period information.

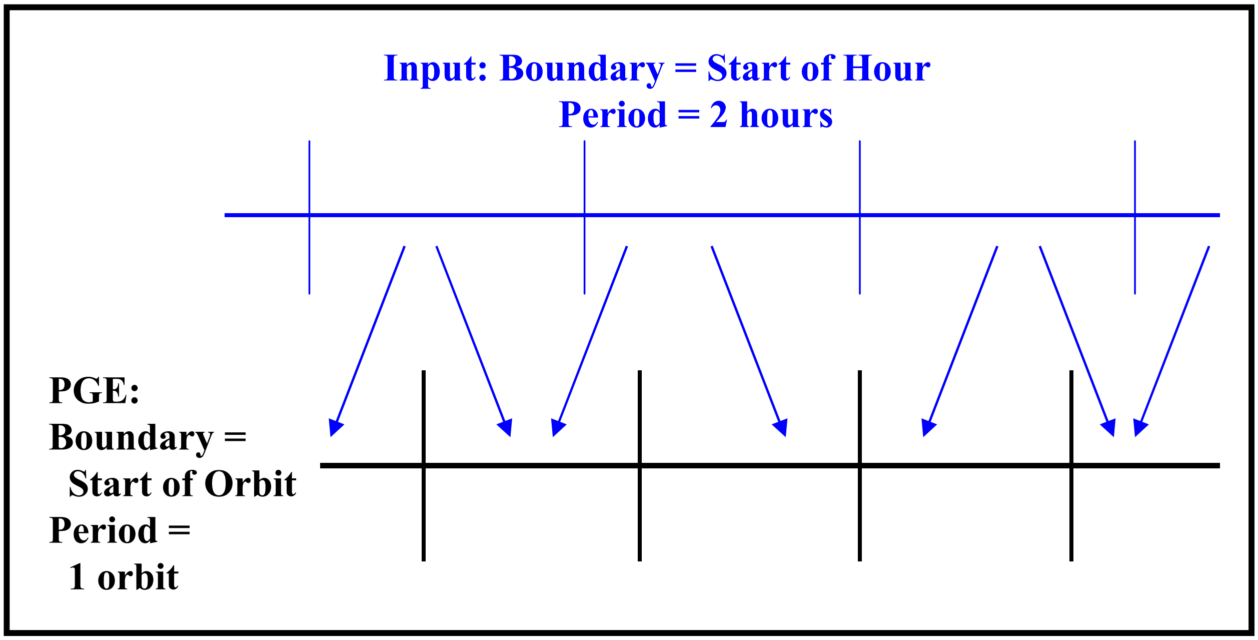

Figure 11.2.9-1. Example of the Orbital Processing Production Rule

11.2.9.1 PGE Science Metadata ODL File Parameters

The following parameters must be set in the PGE science metadata ODL file in order to implement the Orbital Processing Production Rule:

- PLATFORM.
- SCHEDULE_TYPE.
- PROCESSING_PERIOD.
- PROCESSING_BOUNDARY.

The PLATFORM parameter is the name of the platform (satellite) for which the PGE is processing data. Information concerning the orbits of a satellite is stored in the PDPS database. Values that can be assigned to the parameter are subject to the following constraints:

- The string specified can have no more than 25 characters.
- The string specified should match the string specified in the orbit science metadata ODL file. If no matching file is found, an error is reported during SSI&T.

The SCHEDULE_TYPE parameter describes the type of scheduling that is required for the PGE. "Orbit" is the value used for Orbital Processing. As a result, the PGE is scheduled based on the start time and period of the satellite's orbit. Note that PROCESSING_PERIOD and PROCESSING_BOUNDARY must be set correspondingly.

The PROCESSING_PERIOD is the time interval for the data that the PGE processes. Assuming no combination of production rules that would affect the period, data are acquired for the specified PROCESSING_PERIOD and output data are planned for the given period. The value assigned to PROCESSING_PERIOD is of the format "<Period Type>=<Length of Period>".

The “Period Type” applicable to the Orbital Processing Production Rule is "ORBITS". For example, "ORBITS=1" would be applied to a PGE that processes data related to one orbit’s worth of data.

The PROCESSING_BOUNDARY is the boundary (starting point in time) of the PGE. It specifies when each instance of the PGE should start. Note that the PROCESSING_BOUNDARY and PROCESSING_PERIOD are used in conjunction when scheduling the PGE. Consequently, "START_OF_ORBIT" is the acceptable PROCESSING_BOUNDARY for the Orbital Processing Production Rule. It indicates that the PGE processes data related to each satellite orbit. It must be used in conjunction with a PROCESSING_PERIOD that specifies a “Period Type” of “ORBITS”.

11.2.9.2 Orbit Science Metadata ODL File Parameters

The following parameter must be set in the orbit science metadata ODL file in order to implement the Orbital Processing Production Rule:

- PLATFORM.

In addition, the following ODL object is used in defining orbits for the Orbital Processing Production Rule:

- ORBIT_MODEL object.

The value assigned to the PLATFORM parameter in the orbit science metadata ODL file must be exactly the same as that specified for the same parameter in the PGE science metadata ODL file.

The ORBIT_MODEL object is an ODL object that surrounds each orbit definition. An OBJECT/END_OBJECT pair (as shown in the example that follows) is needed for each orbit that is to be expressly defined. PDPS extrapolates or interpolates orbits that are not specifically defined within the file.

```
OBJECT = ORBIT_MODEL
  CLASS = 1
  ORBIT_NUMBER = 1000
  ORBIT_PATH_NUMBER = 68
  ORBIT_PERIOD = "MINS=98"
  ORBIT_ START = “09/21/1999 14:50:00”
END_OBJECT = ORBIT_MODEL
```

The following parameters are set in the ORBIT_MODEL object in order to implement the Orbital Processing Production Rule:

- CLASS.
- ORBIT_NUMBER.
- ORBIT_PATH_NUMBER.
- ORBIT_PERIOD.
- ORBIT_START.

CLASS is a simple counter used to differentiate the different ORBIT_MODEL objects within the file. Each ORBIT_MODEL object needs to have a different CLASS value.

ORBIT_NUMBER is simply the number of the orbit being specified. Each orbit of the satellite has a sequential number associated with it. This is the integer value of the orbit number for the orbit being defined in the ORBIT_MODEL object.

ORBIT_PATH_NUMBER is value of the path for the specified orbit. The orbital path is a number from 0-233 that repeats every 16 days. This is the integer value of the orbital path number for the orbit being defined in the ORBIT_MODEL object.

ORBIT_PERIOD is the length of time it takes for the satellite to complete one orbit. The value assigned to ORBIT_PERIOD has the format "<Period Type>=<Length of Period>" (e.g., "MINS=98"). Note that the "Length of Period" is specified as a positive integer only.

Period Type values for the orbit model science metadata ODL file are:

- "WEEKS"
  - Orbit spans some number of weeks.
  - For example, "WEEKS=2" would be an orbit that takes two weeks to complete.
- "DAYS"
  - Orbit spans some number of days.
  - For example, "DAYS=5" would be an orbit that takes five days to complete.
- "HOURS"
  - Orbit spans some number of hours.
  - For example, "HOURS=4" would be an orbit that takes four hours to complete.
- "MINS"
  - Orbit spans some number of minutes.
  - For example, "MINS=85" would be an orbit that takes eighty-five minutes to complete.
- "SECS"
  - Orbit spans some number of seconds.
  - For example, "SECS=7200" would be an orbit that takes 7200 seconds (two hours) to complete.

ORBIT_START is the start date and time for the orbit defined by the particular ORBIT_MODEL object. Its format is either "MMM DD YYYY HH:MM:SS" or "MM/DD/YYYY HH:MM:SS".

11.2.10 Multiple DPRs for Insertion Time Production Rule

The Multiple DPRs for Insertion Time production rule allows the creation of DPRs for multiple granules with the same insertion time. It affects ASTER L1B routine processing only.

The Multiple DPRs for Insertion Time production rule is implemented when the Production Planner creates a production request using the Production Request Editor. The Production Planner enters the Duration information (Begin and End date and time) for the Insertion Time (versus Collection Time) and then ensures that the **Multiple DPRs** toggle button is depressed. If the **Multiple DPRs** toggle button is not depressed, the default rule is applied and when the production request is saved a DPR is created for the most recent granule with the specified insertion time.

The Multiple DPRs for Insertion Time production rule applies to the key input (e.g., ASTER L1B) only. It does not affect any ancillary data types (for which the most recent granule is selected).

When the Multiple DPRs for Insertion Time production rule is invoked, a DPR is generated for each key-data-type granule returned from the insertion-time query. Therefore, if multiple key-data-type granules are returned for the same collection period, multiple DPRs are generated. The DPR IDs are different in the last three characters, which are generated randomly.

11.2.11 Tiling Production Rule

The Tiling Production Rule allows a PGE to run over a series of specific geographic locations called "tiles". The tiles are defined before the PGE is scheduled, specifying the longitude and latitude of four points that outline each tile. When the PGE is scheduled, it is scheduled for an entire day, and data is queried based on both a timeframe and the geographic location specified. Each run of the PGE for that day is for a specific tile, and only data that overlap or fit within the geographical coordinates of the tile are staged for the PGE.

- Example:
  - A MODIS PGE is designed to run on data for a specific geographic location every day.
  - The location is expressed as a polygon defined by latitude and longitude coordinates.
  - The MODIS PGE is scheduled every day, and data are retrieved that match the time period (the day for which the PGE is being executed) and some part of it falls within the geographic constraints of the tile.
  - The PGE runs and produces data that define information about the particular tile.

Period and **boundary** are used to specify the timing of input data and provide indications of how often the PGE should be executed. But at least some of the input data are retrieved on the basis of the coordinates defined for the tile on which the PGE is executing. In fact there are really two kinds of tiling:

- The PGE takes in data based on geographic shapes (tiles) and produces an output or outputs for the specified geographical coverage.
- The PGE takes in an already tiled product as input.
  - This form of tiling is more like a Metadata Query using a runtime parameter value to acquire the correct tiled data.

There are some possible future enhancements to the Tiling Production Rule but they have not been scheduled yet.

- **Zonal Tiling** supports tiles that cover a band around the Earth between two given latitudes.

- **Tile Clustering** involves grouping tiles that cover nearby geographic locations together so that data that span the tiles may be staged only once.
  - Intended to improve the performance of Tiling.
  - Also provides for the ability to prioritize one group of tiles over others (so specific geographic outputs are produced before other geographic outputs).

Runtime parameters can be set to the ID of the tile being processed. Since PDPS schedules a Tiling PGE to run once per tile, it can pass the identifier of the tile to the PGE. The identifier can be placed under a specified runtime parameter in the PCF, or it can be used in a Metadata Query for a PGE that would use already tiled data as input.

Figure 11.2.11-1 provides an example of the Tiling Production Rule. The PGE runs once per defined tile. So for every tile in the Tile Scheme a Data Processing Request is created to run using data that match the geographic extent of the tile. The PDPS sends the coordinates of the tiles (e.g., Tiles 1 through 3 in Figure 11.2.11-1) to the Science Data Server when requesting data and acquires only the granules that fall fully or partially within the defined tile.

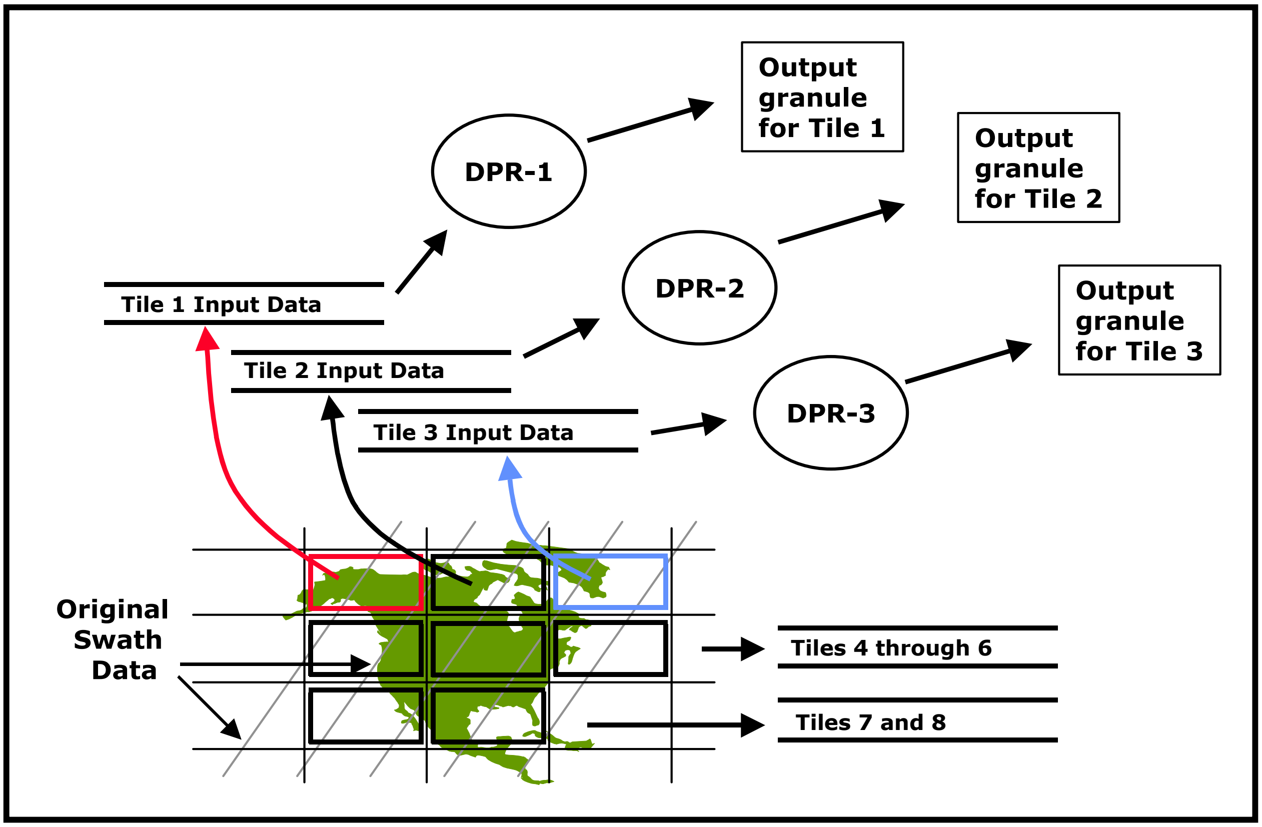

Figure 11.2.11-1. Example of the Tiling Production Rule

The PGE itself must be set up to handle the fact that the entire area of the tile may not be covered by available data. In addition, because PDPS does not keep track of tiles once they have been produced, the PGE must set the metadata of the output products so a downstream Tiling PGE can acquire the correct granules for a given tile. The PDPS matches up the granules needed for a downstream PGE via a query to the Data Server Subsystem.

11.2.11.1 Tiling Based on Already Tiled Data

As previously stated the second form of Tiling concerns PGEs based on tiles that have already been created by other PGEs. Tiling based on already tiled data is really a combination of the Metadata Query Production Rule and the Tiling Production Rule. The latter is used in running the PGE(s) once per tile, just like any other Tiling PGE. The Metadata Query Production Rule is used in acquiring the previously tiled data by querying the Science Data Server for metadata that match the tile ID that is currently being executed. The query depends on the **runtime parameters** function of Tiling to provide the tile ID relevant to the PGE that is currently being executed.

The Tiling Production Rule is based (at least for the PGE science metadata ODL file) on the same fields used for the Basic Temporal Production Rule. A PGE that performs Tiling still needs a **boundary** and **period** and other such parameters. The difference is that values specified for some of the fields provide Tiling information. Furthermore, Tiling requires that a tile scheme be identified in the PGE science metadata ODL file. The tile scheme is defined in a tile science metadata ODL file.

11.2.11.2 PGE Science Metadata ODL File Parameters

The following parameters must be set in the PGE science metadata ODL file in order to implement the Tiling Production Rule:

- SCHEDULE_TYPE.
- TILE_SCHEME_NAME.

In addition, the following parameter is used within a PCF_ENTRY when defining the Tiling Production Rule:

- QUERY_TYPE.

The SCHEDULE_TYPE parameter defines the type of scheduling that will be done for the PGE. Values for the Tiling Production Rule are:

- "Tiling"
  - Tile-Scheduled.
  - The PGE is scheduled based on the specified PROCESSING_PERIOD and PROCESSING_BOUNDARY, but a DPR is created for each defined tile.

The TILE_SCHEME_NAME parameter is the name of the Tile Scheme to be used by PDPS when scheduling and executing PGEs for each defined tile. There must be a tile ODL file that matches the specified scheme name.

The QUERY_TYPE parameter specifies the type of query to be performed on the input defined by the PCF_ENTRY Object. It uses the following syntax:

```
OBJECT = PCF_ENTRY
  .
  .
  .
  QUERY_TYPE =
  .
END_OBJECT = PCF_ENTRY
```

For Tiling PGEs there are two possible values for QUERY_TYPE:

- "Tile"
  - The data for the input are acquired on the basis of the spatial constraints of the current tile.
  - Used for a PGE that takes in raw data and produces one or more tiles of data.
- "Already Created Tile"
  - The input is a tiled output of another Tiling PGE.
  - Used for a PGE that takes input from one or more other Tiling PGEs.
  - A Metadata Query must be added to this PCF_ENTRY in order for the correct tiled input to be acquired.

11.2.11.3 Tile Science Metadata ODL File Parameters

The following parameter must be set in the Tile science metadata ODL file in order to implement the Tiling Production Rule:

- TILE_SCHEME_NAME.

In addition, the following ODL objects are used within a PCF_ENTRY to define the Tiling Production Rule:

- TILE object.
- TILE_COORDINATE object.

The TILE_SCHEME_NAME parameter identifies the tile scheme for which the tile information is being specified. Values are limited by the following constraints:

- The string specified can be no more than 20 characters.
- The string specified should match the string specified for TILE_SCHEME in the PGE science metadata ODL file.

The TILE object is an ODL object that surrounds each tile definition. An OBJECT/END_OBJECT pair (as shown in the example that follows) is needed for each tile that is going to be expressly defined:

```
OBJECT = TILE
  .
  .
```

```
            .
            END_OBJECT = TILE
```

The following parameters are set in the TILE object in order to implement the Tiling Production Rule:

- CLASS.
- TILE_ID.
- TILE_DESCRIPTION.

CLASS is a simple counter used to differentiate the different TILE objects within the file. Each TILE object needs to have a different CLASS value.

TILE_ID is the tile identifier for the tile being defined. The TILE_ID must be an integer (e.g., TILE_ID = 12) and must be greater than zero but less than the maximum integer. If a Tile ID is defined in other tile schemes, it must have the same coordinates and description.

TILE_DESCRIPTION is a string of characters (255 characters maximum) that describes what the tile is for, such as its geographic location or area that it covers (e.g., TILE_DESCRIPTION = "Upper North America").

The TILE_COORDINATE object is an ODL object that defines a coordinate (latitude and longitude) for a tile. An OBJECT/END_OBJECT pair is needed for each coordinate that is defined. Each tile must have four TILE_COORDINATE objects defined. (Currently only four-sided polygons are allowed; however, a possible future enhancement would provide for polygons with more than four points.) Coordinate objects must follow a clockwise sequence so that if lines were drawn between the points in the order they are given the desired shape would be drawn.

Coordinate objects conform to the following format:

```
OBJECT = TILE
  .
  .
  .
  OBJECT = TILE_COORDINATE
    CLASS =
    LATITUDE =
    LONGITUDE =
  END_OBJECT = TILE_COORDINATE
  .
  .
  .
END_OBJECT = TILE
```

The following parameters are set in the TILE_COORDINATE object in order to implement the Tiling Production Rule:

- CLASS.
- LATITUDE.

- LONGITUDE.

The CLASS parameter (e.g., CLASS = 1) is an object counter that is used only to distinguish objects. The value assigned to CLASS must be an integer greater than zero and must be unique in the file for the particular type of object.

The LATITUDE parameter (e.g., LATITUDE = 12.15) describes the latitude component of the tile coordinate. There is one LATITUDE entry per TILE_COORDINATE object.

The LONGITUDE parameter (e.g., LONGITUDE = -43.22) describes the longitude component of the tile coordinate. There is one LONGITUDE entry per TILE_COORDINATE object.

11.3 Intermittent Activation

The conditions for executing most PGEs are well defined. The most common activation condition is the availability of all input data sets. Similarly, the frequency of execution is usually well defined (e.g., run once for every granule or run monthly averages once a month). However, some PGEs have additional or different constraints on when they are run.

A PGE can be set up to run on every n^{th} instance of input data. For example, a QA PGE that is run on a daily product may need to be run only every fifth day to provide a spot check. Note that this does **not** refer to the common case of running a weekly averaging PGE only once each week, which would be handled by the Basic Temporal Production Rule and the time ranges specified for the input and output ESDTs. Rather, this is a special case where a PGE **can** be run every day (or hour, week, etc.), but for some reason (such as a QA check) it is desired to run the PGE only every n^{th} day.

To implement Intermittent Activation the Production Planner supplies the following information (via the Production Request Editor) when creating a production request:

- **Number to Skip**
  - Number of DPRs to be skipped (not executed).
  - Entered in the **Skip** field on the Production Request Editor.
- **Number to Keep**
  - After skipping the specified number of DPRs, how many are to be kept?
  - Entered in the **Keep** field on the Production Request Editor.
  - The number to keep is usually one but could be any number.
- **Skip First**
  - Button on the Production Request Editor.
  - Selected to skip the first DPR.
  - Not selected if the first DPR is to be run.

The Planning Subsystem uses the preceding information to establish a pattern of execution. The pattern is effective for the single PR in which the “number to skip” and the “number to keep” are specified; it is not maintained between PRs.

The following example of Intermittent Activation is illustrated in Figure 11.3-1:

- The Production Planner prepares a production request for a 14-day period, generating 14 DPRs.
- The Production Planner made the following selections on the Production Request Editor:
  - Entered “4” in the **Number to Skip** field.
  - Entered “1” in the **Number to Keep** field.
  - Did **not** select the **Skip First** button.
- Consequently, the following results are obtained:
  - First DPR runs.
  - Four DPRs (second through fifth) are skipped.
  - Sixth DPR runs.
  - Four DPRs (seventh through tenth) are skipped.
  - Eleventh DPR runs.
  - Remaining three DPRs (twelfth through fourteenth) are skipped.

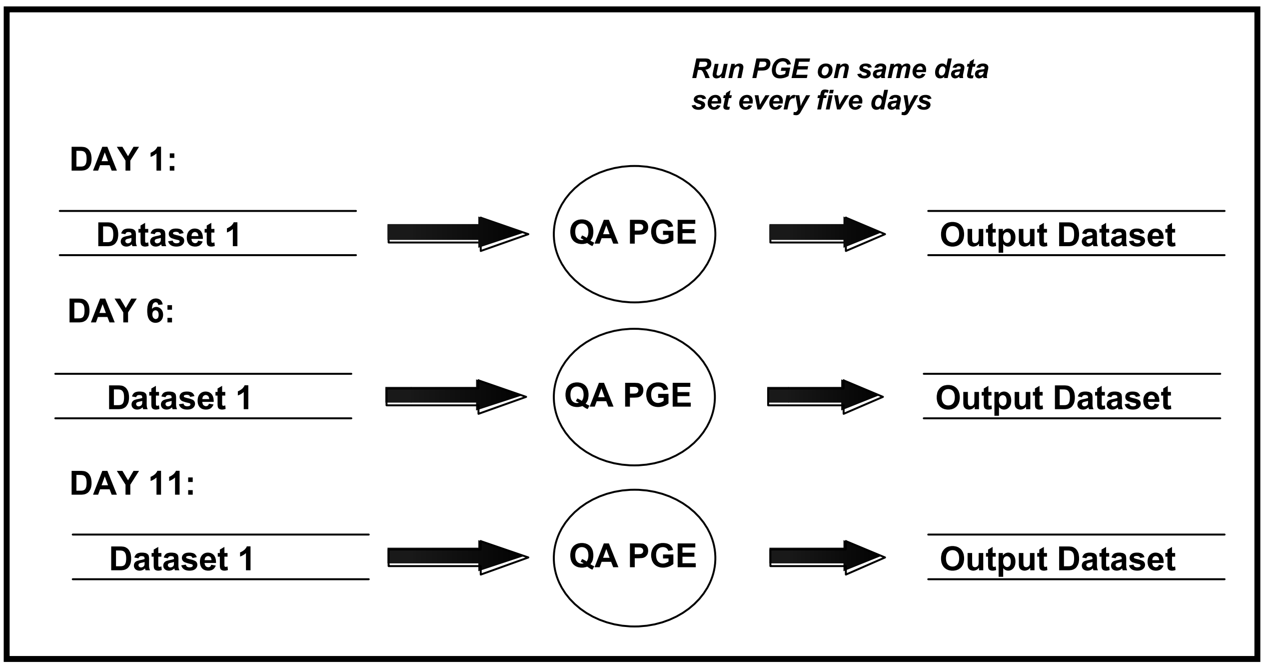

Figure 11.3-1. Example of Intermittent Activation

12. Resource Planning

12.1 Resource Planning Process

The Resource Planning process is the mechanism by which reservations for non-routine ground events are defined and controlled. Such events may include testing, corrective maintenance, preventive maintenance or system upgrades, or any other event that requires DAAC production processing resources. Resource planning defines ground events, which are also used in production planning; thus, resource planning can take place whenever a production plan needs to be created. In general, this will occur on a biweekly basis for 30-day plans, on a weekly basis for ten-day plans, and on a daily basis. However, ground events can be entered at any time. The important point is that it is necessary to be aware of the anticipated processing load and upcoming maintenance events for about the next month.

Resource Planning includes two general types of activities; i.e., Resource Definition and Resource Scheduling. The site Resource Planner uses the Resource Editor GUI within the Planning Subsystem to define ECS resources used in production processing. The Resource Planner and Resource Manager use the Resource Scheduler GUI to schedule non-routine events against ECS resources.

Subsequent sections related to Resource Planning address the following topics:

- Section 12.2 An overview of the process for defining resources and step-by-step procedures for using the Resource Editor.
- Section 12.3 An overview of the process for scheduling resources and step-by-step procedures for using the Resource Scheduler.
- Section 12.4 An overview of the process for tuning system parameters related to Resource Planning and a description of the process for changing configuration parameters.
- Section 12.5 An overview of the process and step-by-step procedures for troubleshooting Resource Planning problems.

12.2 Defining Resources

The Resource Planner uses the Resource Editor GUI to define ECS resources used in production processing in the following terms:

- "Disks."
- "Virtual computers" (sets of CPUs and associated memory and disks).
- "Strings" (sets of virtual computers).
- "Real computers" (hosts that are composed of one or more virtual computers).
- "AutoSys" ("strings" associated with the production processing software).
- Generic "hardware."

The following general process is used for defining production resources:

- Determine what production resources are available.
- Determine the distribution of resources among operating modes.
- Define resources for each mode using the Resource Editor GUI.

Each procedure outlined has an **Activity Checklist** table that provides an overview of the task to be completed. The outline of the **Activity Checklist** is as follows:

Column one - ***Order*** shows the order in which tasks could be accomplished.

Column two - ***Role*** lists the Role/Manager/Operator responsible for performing the task.

Column three -***Task*** provides a brief explanation of the task.

Column four - ***Section*** provides the Procedure (P) section number or Instruction (I) section number where details for performing the task can be found.

Column five - **Complete?** is used as a checklist to keep track of which task steps have been completed.

Table 12.2-1, below, provides an Activity Checklist for Defining Resources.

Table 12.2-1. Defining Resources - Activity Checklist

Order	Role	Task	Section	Complete?
1	Resource Planner	Log in to ECS Hosts	(P) 12.2.1	
2	Resource Planner	Launch the Resource Editor	(P) 12.2.2	
3	Resource Planner	Determine Actual Processing Resources	(P) 12.2.3	
4	Resource Planner	Add a Resource	(P) 12.2.4	
5	Resource Planner	Add/Modify a Disk	(P) 12.2.4.1	
6	Resource Planner	Add/Modify a Virtual Computer	(P) 12.2.4.2	
7	Resource Planner	Add/Modify a Real Computer	(P) 12.2.4.3	
8	Resource Planner	Add/Modify a String	(P) 12.2.4.4	
9	Resource Planner	Add/Modify an Autosys Resource	(P) 12.2.4.5	
10	Resource Planner	Add/Modify a Hardware Resource	(P) 12.2.4.6	
11	Resource Planner	Modify a Resource	(P) 12.2.5	
12	Resource Planner	Delete a Resource	(P) 12.2.6	
13	Resource Planner or DAAC Staff	Shut Down Resource Definition Applications	(P) 12.2.7	

12.2.1 Log in to ECS Hosts

Logging in to ECS hosts is accomplished from a UNIX command line prompt. Table 12.2-2 presents (in a condensed format) the steps required to log in to ECS hosts. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1 At the UNIX command line prompt enter:

setenv DISPLAY <client name>:0.0

- Use either the X terminal/workstation IP address or the machine-name for the client name.
- When using secure shell, the DISPLAY variable is set just once, before logging in to remote hosts. If it were to be reset after logging in to a remote host, the security features would be compromised.

2 In the terminal window (at the command line prompt) start the log-in to the appropriate host by entering:

/tools/bin/ssh <host name>

- The **-l** option can be used with the ssh command to allow logging in to the remote host (or the local host for that matter) with a different user ID. For example, to log in to x0pls02 as user cmops enter:

  /tools/bin/ssh -l cmops x0pls02

- Depending on the set-up it may or may not be necessary to include the path (i.e., /tools/bin/) with the ssh command. Using ssh alone is often adequate. For example:

  ssh x0pls02

  - or -

  ssh -l cmops x0pls02

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- Examples of Science Processor host names include **e0spg11**, **g0spg11,** and **l0spg11**.
- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03**.
- Examples of Sun internal server host names include **e0acs06**, **g0acs06**, and **l0acs06**.
- Examples of Access/Process Coordinators (APC) Server host names include **e0acg11**, **g0acg01**, and **l0acg02**.
- Examples of Ingest Server host names include **e0icg11, g0icg01**, and **l0acg02**.
- Examples of Sun external server host names include **e0ins01, g0ins01**, and **l0ins01**.

- If you receive the message, "Host key not found from the list of known hosts. Are you sure you want to continue connecting (yes/no)?" enter **yes** ("y" alone will not work).
- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter *passphrase* for RSA key '<*user@localhost*>'** appears; continue with Step 3.
- If you have not previously set up a secure shell passphrase, go to Step 4.

3 If a prompt to **Enter passphrase for RSA key '<*user@localhost*>'** appears, enter:

<passphrase>

- If a command line prompt is displayed, log-in is complete.
- If the passphrase is unknown, press **Return/Enter**, which should cause a **<user@remotehost>'s password:** prompt to appear (after the second or third try if not after the first one), then go to Step 4.
- If the passphrase is entered improperly, a **<user@remotehost>'s password:** prompt should appear (after the second or third try if not after the first one); go to Step 4.

4 If a prompt for **<*user@remotehost*>'s password:** appears, enter:

<password>

- A command line prompt is displayed.
- Log-in is complete.

Table 12.2-2. Log in to ECS Hosts - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**setenv DISPLAY <client name>:0.0**	**enter text, press Enter**
2	**/tools/bin/ssh <host name>** (as applicable)	**enter text, press Enter**
3	**<passphrase>** (if applicable)	**enter text, press Enter**
4	**<password>** (if applicable)	**enter text, press Enter**

12.2.2 Launch the Resource Editor

The Resource Editor is invoked from a UNIX command line prompt. Table 12.2-3 presents (in a condensed format) the steps required to launch the **Resource Editor** GUI. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1 Access a terminal window logged in to the Planning/Management Workstation host.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 12.2.1).

2 In the terminal window, at the command line, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1 - Science Software Integration and Test (SSI&T)
  - TS2 - New Version Checkout
  - OPS - Normal Operations
- "utilities" is the directory containing the Planning Subsystem start-up scripts.

3 Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered.
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

4 Start the Resource Planning background processes by entering:

EcPlRpAllStart <MODE> <application_id>

- The Resource Planning background processes are launched.
- The **application_id** or **MSGSRV_ID** is a number from 1 to 5. It identifies the message service in use so messages can be directed to the proper message handler GUI. Consequently, it is a good idea to use the same application_id consistently during a resource planning session.

5 Start the **Resource Editor** GUI by entering:

EcPlRpReStart <MODE> <application_id>

- The **Resource Editor** GUI is launched.
- The **Resource Editor** GUI displays a list of defined resources and a series of buttons that enable the following operations:
  - **New...** Add a resource definition. (Section 12.2.4)
  - **Modify...** Edit or review the details of an existing resource definition. (Section 12.2.5)
  - **Delete** Delete a resource definition. (Section 12.2.6)
  - **Fetch Baseline** [not used]
  - **Load Baseline** [not used]

Table 12.2-3. Launch the Resource Editor - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 12.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	Set the environment variables	**enter text, press Enter**
4	**EcPlRpAllStart <MODE> <application_id>**	**enter text, press Enter**
5	**EcPlRpReStart <MODE> <application_id>**	**enter text, press Enter**

12.2.3 Determine Actual Processing Resources

The Resource Editor allows the authorized operator to define resources in the following categories:

- **Disks:** Disk partitions that are associated with and provide temporary data storage for the input and output files used in processing.
- **Virtual Computers:** Virtual computers composed of CPUs, random-access memory (RAM), and associated-disk(s). The CPUs and RAM specified for a virtual computer are components of the real computer from which the virtual computer is derived.
- **Strings:** Sets of one or more virtual computers. Strings are associated with the processing software (AutoSys). A dual science processor configuration can be defined by specifying strings containing virtual computers derived from different real computers.
- **Real Computers:** Physical computing devices (hosts), each of which contains one or more CPUs. Each science processor host ("real" computer) is divided into one or more virtual computers by allocating CPUs and RAM from the real computer to the virtual computer(s).
- **AutoSys:** Identifies the string(s) of virtual computers used by the production processing software.
- **Hardware:** Any type of equipment that is not defined as a computer or disk may be defined as "hardware."

The ECS Operational Readiness Plan for Release 2.0 (603-CD-003-001) specifies that initially disk partitions at the DAACs are to be split among the operating modes as follows:

- OPS – 60%.
- TS1 - 20%.
- TS2 - 20%.

However, it may be advantageous to reserve some nominal percentage of the disk (e.g., two to five percent) as a safety buffer. In any case, it is critical to ensure that the sum of the disk space assigned to the various modes is no more than the total disk space available.

Although the ECS Operational Readiness Plan does not specifically mention allocating resources other than disk partitions, CPUs and RAM need to be allocated among modes in the same manner. However, it is not necessary to be exact with the CPU count or RAM amount.

- There is no one-to-one mapping of CPU allocation with actual CPUs on the science processor.
- Actual CPU usage during processing is limited by the operating system (OS).
  - If ten CPUs have been specified for a particular mode, only ten Data Processing Requests (DPRs) can be running the Execute job at a given time.
  - What is really being defined is the maximum number of DPRs that will execute at a given time.
- It is important to monitor the load on each science processor.
  - CPUs can be over-allocated or under-allocated as necessary to get the most out of the CPUs on each science processor.
  - If monitoring indicates that the processor is underused when OPS mode is at full processing capacity, the number of CPUs allocated to OPS mode could probably be increased.
  - If the science processor is at full capacity when OPS mode is at full processing capacity and it is suspected that the processor may be overworked, the number of CPUs allocated to OPS mode should be reduced.
- Random-access memory (RAM) is subject to the same considerations as CPUs.
  - RAM can be over-allocated or under-allocated as necessary to get the most out of the memory on each science processor.
- The OS takes care of true CPU and RAM allocation.

Before data processing resources can be defined, it is necessary to know what resources are actually available at the DAAC. Some resources are defined in terms of other resources; for example, a string is defined as one or more virtual computers. However, it is generally necessary to have the following types of information available in order to define processing resources:

- Host names [“real computers”].
- Number of processors [CPUs] available on each host.
- Operating System (OS) for each host.
- Memory [RAM] on each host.
- Total disk space.
- AutoSys instance(s) at the DAAC.

Table 12.2-4 presents (in a condensed format) the steps required to determine the actual processing resources to be defined. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

NOTE: The procedure to determine the actual processing resources to be defined starts with the assumption that the DISPLAY environment variable has been set (Refer to Section 12.2.2).

1 Access a terminal window logged in to the applicable Science Processor host.

- Examples of Science Processor host names include **e0spg11**, **g0spg11,** and **l0spg11**.
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 12.2.1).

2 To access the mount point enter:

cd /usr/ecs/<MODE>/CUSTOM/pdps/<processor>/data/DpPrRm/<processor>_disk

- Change directory to the disk mount point (e.g., /usr/ecs/OPS/CUSTOM/pdps/e0spg11/data/DpPrRm/e0spg11_disk).

3 To determine disk size and usage enter:

df -k . (being sure to include the dot)

- Information concerning disk size and use is displayed; for example:

  Filesystem Type kbytes use avail %use Mounted on
  /dev/dsk/xlv/vgo1 xfs 413394688 164646048 248748640 40 /vol1

- In the preceding example the total disk space is 413,394,688 kilobytes or 413,394.69 megabytes (413 gigabytes).

4 To obtain Information concerning the number of CPUs and amount of RAM (memory) enter:

hinv

- The hinv command is available on Silicon Graphics, Inc. (SGI) hosts only.
- Information concerning CPUs and RAM (memory) is displayed; for example (not all rows are shown):

  Processor 0: 194 MHZ IP25
  CPU: MIPS R10000 Processor Chip Revision: 2.6
  FPU: MIPS R10010 Floating Point Chip Revision: 0.0
  Processor 1: 194 MHZ IP25
  CPU: MIPS R10000 Processor Chip Revision: 2.6
  FPU: MIPS R10010 Floating Point Chip Revision: 0.0
  Processor 2: 194 MHZ IP25
  CPU: MIPS R10000 Processor Chip Revision: 2.6
  FPU: MIPS R10010 Floating Point Chip Revision: 0.0
  Processor 3: 194 MHZ IP25
  CPU: MIPS R10000 Processor Chip Revision: 2.6
  FPU: MIPS R10010 Floating Point Chip Revision: 0.0
  Processor 4: 194 MHZ IP25
  CPU: MIPS R10000 Processor Chip Revision: 2.5
  FPU: MIPS R10010 Floating Point Chip Revision: 0.0

Processor 5: 194 MHZ IP25
CPU: MIPS R10000 Processor Chip Revision: 2.5
FPU: MIPS R10010 Floating Point Chip Revision: 0.0
Processor 6: 194 MHZ IP25
CPU: MIPS R10000 Processor Chip Revision: 2.5
FPU: MIPS R10010 Floating Point Chip Revision: 0.0
Processor 7: 194 MHZ IP25
CPU: MIPS R10000 Processor Chip Revision: 2.5
FPU: MIPS R10010 Floating Point Chip Revision: 0.0
Processor 8: 194 MHZ IP25
CPU: MIPS R10000 Processor Chip Revision: 2.5
FPU: MIPS R10010 Floating Point Chip Revision: 0.0
Processor 9: 194 MHZ IP25
CPU: MIPS R10000 Processor Chip Revision: 2.5
FPU: MIPS R10010 Floating Point Chip Revision: 0.0
Processor 10: 194 MHZ IP25
CPU: MIPS R10000 Processor Chip Revision: 2.5
FPU: MIPS R10010 Floating Point Chip Revision: 0.0
Processor 11: 194 MHZ IP25
CPU: MIPS R10000 Processor Chip Revision: 2.5
FPU: MIPS R10010 Floating Point Chip Revision: 0.0
Processor 12: 194 MHZ IP25
CPU: MIPS R10000 Processor Chip Revision: 2.5
FPU: MIPS R10010 Floating Point Chip Revision: 0.0
Processor 13: 194 MHZ IP25
CPU: MIPS R10000 Processor Chip Revision: 2.5
FPU: MIPS R10010 Floating Point Chip Revision: 0.0
Processor 14: 194 MHZ IP25
CPU: MIPS R10000 Processor Chip Revision: 2.5
FPU: MIPS R10010 Floating Point Chip Revision: 0.0
Processor 15: 194 MHZ IP25
CPU: MIPS R10000 Processor Chip Revision: 2.5
FPU: MIPS R10010 Floating Point Chip Revision: 0.0
Secondary unified instruction/data cache size: 1 Mbyte
Data cache size: 32 Kbytes
Instruction cache size: 32 Kbytes
Main memory size: 2048 Mbytes, 8-way interleaved
[…]

- In the example the science processor has 16 CPUs (Processor 0 – Processor 15) and 2048 megabytes of RAM.

5 Repeat Steps 1 through 4 for all other science processors (if any).

NOTE: Steps 6 through 12 describe the use of the Netscape browser to determine certain types of information concerning computer resources (including the number of CPUs and amount of RAM), which can be determined using the **hinv** command as described in Step 4. However, the "as-built" file accessed using the Netscape browser lists the necessary operating system information in addition to CPU and RAM data. The advantage of the **hinv** command is that it provides real-time data and is reliably up to date. The advantage of the "as-built" file accessed using the Netscape browser is that it provides operating system data that is not available using the **hinv** command.

6 To launch the Netscape web browser enter:

netscape &

- It may be necessary to change directories before launching the Netscape web browser (e.g., cd /tools/bin/netscape3.01).
- The Netscape web browser is displayed.

7 In the browser's **Location** (**Go To**) field enter the following address:

http://cmdm.east.hitc.com/baseline

- The ECS Baseline Information System web page is displayed.

8 **Single-click** on the **ECS Configuration** link.

- A table of files is displayed.

9 **Single-click** on the **Asbuilts** link for the relevant DAAC.

- A list of files is displayed.

10 **Single-click** on the file name corresponding to the desired host.

- For example, the file name for host x0spg01 would be x0spg01.asbuilt.html.
- A report containing the following types of information (among other items) is displayed:
  - **Host Name** ["real computer"].
  - **Processors** [CPUs].
  - **Operating System.**
  - **Memory** [RAM].
  - **Interrogation Date** (useful in determining how up-to-date the information is).

11 **Single-click** on the browser **Back** button.

- The list of "as-built" files is displayed.

12 Repeat Steps 10 and 11 for all other science processors (if any).

13 Exit from the Netscape browser when the necessary information has been acquired by executing the following menu path:

File → Exit

- The Netscape browser disappears.

14 Access a terminal window logged in to the Queuing Server host.

- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03**.
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 12.2.1).

15 To gain access to the directory containing the AutoSys configuration files enter:

cd /usr/ecs/<MODE>/COTS/<autotree>/autouser

- Change directory to the directory (e.g., **/usr/ecs/<MODE>/COTS/autotreeb/autouser**, **/usr/ecs/<MODE>/COTS/autotree/autouser**, **/data1/SHARED/COTS/autotree/autouser**) containing the set-up files (e.g., **FMR.autosys.csh.g0sps06**) and the AutoSys configuration files (e.g., **config.FMR**).
- The particular path to be typed may vary from site to site.
- The AutoSys instance at the DAAC is identified by three capital letters appended to the beginning of the set-up files and the end of the configuration file.
  - Typically, AutoSys instances at the DAACs are identified as **FMR**.
- It is possible to have multiple AutoSys instances installed at a DAAC.

Table 12.2-4. Determine Actual Processing Resources - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	UNIX window (Science Processor)	**single-click** or use procedure in Section 12.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/pdps/<processor>/data/DpPrRm/<processor>_disk**	**enter text, press Enter**
3	**df -k .** (being sure to include the dot)	**enter text, press Enter**
4	Observe the disk capacity	**read text**
5	**hinv**	**enter text, press Enter**
6	Observe the number of CPUs and total memory (RAM)	**read text**
7	Repeat Steps 1 through 6 for all other science processors (if any)	

Table 12.2-4. Determine Actual Processing Resources - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
8	Launch Netscape	**enter text, press Enter**
9	**http://pete.hitc.com/baseline**	**enter text, press Enter**
10	**ECS Configuration** link	**single-click**
11	**Asbuilts** link (for the relevant DAAC)	**single-click**
12	**<file name>** (corresponding to the desired host)	**single-click**
13	Observe the number of CPUs, total memory (RAM), and Operating System identification	**read text**
14	**Back** button	**single-click**
15	Repeat Steps 12 through 14 for all other science processors (if any)	
16	**File** → **Exit** (to exit from Netscape)	**single-click**
17	UNIX window (Queuing Server)	**single-click** or use procedure in Section 12.2.1
18	**cd /usr/ecs/<MODE>/COTS/<autotree>/autouser**	**enter text, press Enter**
19	Observe the identification of the AutoSys instance	**read text**

12.2.4 Add a Resource

These procedures address adding such resources as computers, disk partitions, strings, and generic hardware resources to the resource planning list. The ECS Operational Readiness Plan for Release 2.0 (603-CD-003-001) specifies that initially disk partitions at the DAACs are to be split among the operating modes as follows:

- OPS – 60%.
- TS1 - 20%.
- TS-2 - 20%.

However, it may be advantageous to reserve some nominal percentage of the disk (e.g., two to five percent) as a safety buffer. In any case, it is critical to ensure that the sum of the disk space assigned to the various modes is no more than the total disk space available.

Table 12.2-5 presents (in a condensed format) the steps required to add a resource. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 If necessary, launch the **Resource Editor** GUI (refer to Section 12.2.2).
- The **Resource Editor** GUI is displayed.

2 From the **Resource Editor** GUI, select the type of resource to be added from the list on the **Resource Type** button.

3 Either **single-click** on the **New…** button or execute the following menu path:

File → New

4 Define the resource as specified in the corresponding procedure section.

- Refer to the specified section for defining the desired type(s) of resources:
  - **Disk** – Section 12.2.4.1.
  - **Virtual Computer** – Section 12.2.4.2.
  - **Real Computer** – Section 12.2.4.3.
  - **String** – Section 12.2.4.4.
  - **Autosys** – Section 12.2.4.5.
  - **Hardware** – Section 12.2.4.6.
- Resources should generally be added in the preceding order (due to dependencies among resources).

5 After the data have been entered, **single-click** on one of the following buttons:

- **Save** to save **and** exit.
- **Cancel** to exit without saving.

Table 12.2-5. Add a Resource - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Editor** GUI (if necessary)	Use procedure in Section 12.2.2
2	**Resource Type** option button	**single-click**
3	**New…** button	**single-click**
4	Make entries in the necessary fields	Use procedures in Sections 12.2.4.1 through 12.2.4.6
5	**Save** button	**single-click**

12.2.4.1 Add/Modify a Disk

1 Enter the relevant information in the following fields on the **Disk Details** GUI:

- **Resource Name** Operator-defined name for the resource. (required)
  - Example: e0spg11_disk_OPS.
- **Activity** System-generated default activity; can be changed by clicking on the bar in the **Activity** field and then clicking on one of the available options.
- **Partition Size** The size of the disk partition, in **megabytes**. (required)
  - Although the label on the GUI implies that partition size should be entered in "blocks," the label is erroneous. Enter the partition size in megabytes.

- **Block Size** Block size in bytes (always 1024) used for the disk. (required)
- **Comments** Operator comments on the resource.

2 After the data have been entered, **single-click** on one of the following buttons:
- **Save** to save **and** exit**.**
- **Cancel** to exit **without** saving.

12.2.4.2 Add/Modify a Virtual Computer

1 Enter the relevant information in the following fields on the **Virtual Computer Details** GUI:
- **Resource Name** Operator-defined name for the virtual computer. (required)
  - Example: e0spg11_vc_OPS.
- **Activity** System-generated default activity; can be changed by clicking on the bar in the Activity field and then clicking on one of the available options.
- **Number of CPUs** Number of CPUs within the virtual computer. (required)
- **Total RAM** The total memory for the virtual computer in megabytes. (required)
- **Operating System** The operating system name/version for the computer. (required)
- **Disks** A list of the disks previously defined for that site. This list of disks from which to select is used when a disk is associated (or disassociated) with the computer. After items are highlighted, arrow buttons will move items from this list to **Associated Disks** or from the list of **Associated Disks** to the **Disk** list.
- **Associated Disks** Disks in this list are associated with the computer.
- **Comments** Operator comments on the resource.

2 After the data have been entered, **single-click** on one of the following buttons:
- **Save** to save **and** exit**.**
- **Cancel** to exit **without** saving.

12.2.4.3 Add/Modify a Real Computer

1 Enter the relevant information in the following fields on the **Real Computer Details** GUI:
- **Resource Name** Operator-defined name for the real resource. (required)
  - Example: e0spg11.
- **Activity** System-generated default activity; can be changed by clicking on the bar in the Activity field and then clicking on one of the available options.
- **Computers** A list of the virtual computers previously defined for that site. This list of virtual computers from which to select is used when a virtual computer is associated (or disassociated) with the real computer. After items are highlighted,

arrow buttons will move items from this list to **Associated Computers** or from the list of **Associated Computers** to the **Computers** list.

- **Associated Computers** Virtual computers in this list are associated with the real computer.
- **Comments** Operator comments on the resource.

2 After data have been entered, **single-click** on the appropriate button from the following selections:

- **Save** to save **and** exit**.**
- **Cancel** to exit **without** saving.

12.2.4.4 Add/Modify a String

1 Enter the relevant information in the following fields on the **String Details** GUI:

- **Resource Name** Operator-defined name for the resource. (required)
  - Example: e0spg11_string_OPS.
- **Activity** System-generated default activity; can be changed by clicking on the bar in the Activity field and then clicking on one of the available options.
- **Computers** A list of virtual computers previously defined for that site. This list of computers from which to select is used when a computer is associated (or disassociated) with the string. After items are highlighted, arrow buttons will move items from this list to **Associated Computers** or from the list of **Associated Computers** to the **Computer** list.
- **Associated Computers** Virtual computers in this list are associated with the string.
- **Comments** Operator comments on the resource.

2 After data have been entered, **single-click** on the appropriate button from the following selections:

- **Save** to save **and** exit**.**
- **Cancel** to exit **without** saving.

12.2.4.5 Add/Modify an AutoSys Resource

1 Enter the relevant information in the following fields on the **Autosys Details** GUI:

- **Resource Name** Operator-defined name for the AutoSys resource. (required)
  - Example: FMR.
- **Activity** System-generated default activity; can be changed by clicking on the bar in the Activity field and then clicking on one of the available options.
- **Strings** A list of the strings previously defined for that site. This list of strings from which to select is used when a string is associated (or disassociated) with the AutoSys resource. After items are highlighted, arrow buttons will move items from

this list to **Associated Strings** or from the list of **Associated Strings** to the **Strings** list.
- **Associated Strings** Strings in this list are associated with the AutoSys resource.
- **Comments** Operator comments on the resource.

2 After data have been entered, **single-click** on the appropriate button from the following selections:
- **Save** to save **and** exit**.**
- **Cancel** to exit **without** saving.

12.2.4.6 Add/Modify a Hardware Resource

1 Enter the relevant information in the following fields on the **Hardware Resource Details** GUI:
- **Resource Name** Operator-defined name for the resource. (required)
  - Example: e0spg11_cdrom_OPS.
- **Activity** System-generated default activity; can be changed by clicking on the bar in the Activity field and then clicking on one of the available options.
- **Comments** Operator comments on the resource.

2 After data have been entered, **single-click** on the appropriate button from the following selections:
- **Save** to save **and** exit**.**
- **Cancel** to exit **without** saving.

12.2.5 Modify a Resource

Table 12.2-6 presents (in a condensed format) the steps required to modify a resource. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 If necessary, launch the **Resource Editor** GUI (refer to Section 12.2.2).
- The **Resource Editor** GUI is displayed.

2 From the list of resources displayed on the **Resource Editor** GUI, **single-click** on the resource to be modified.

3 **Single-click** on the **Modify…** button to access the appropriate Details GUI.

4 Make the modifications.
- For field descriptions, refer to Sections 12.2.4.1 through 12.2.4.6.

5 After data have been entered, **single-click** on the appropriate button from the following selections:

- **Save** to save **and** exit**.**
- Cancel to exit without saving.

Table 12.2-6. Modify a Resource - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Editor** GUI (if necessary)	Use procedure in Section 12.2.2
2	Select the resource to be modified	**single-click**
3	**Modify…** button	**single-click**
4	Make the modifications	Use procedures in Sections 12.2.4.1 through 12.2.4.6
5	**Save** button	**single-click**

12.2.6 Delete a Resource

Table 12.2-7 presents (in a condensed format) the steps required to delete a resource. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 If necessary, launch the **Resource Editor** GUI (refer to Section 12.2.2).

- The **Resource Editor** GUI is displayed.

2 From the list of resources displayed on the **Resource Editor** GUI, **single-click** on the resource to be deleted.

3 **Single-click** on the **Delete** button.

- A dialogue box pops up to verify whether the resource is really to be deleted.

4 **Single-click** on one of the following buttons as appropriate:

- **OK** to remove the resource from the list and from the PDPS database **and** exit**.**
- **Cancel** to exit **without** deleting the resource.

Table 12.2-7. Delete a Resource - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Editor** GUI (if necessary)	Use procedure in Section 12.2.2
2	**<resource>** (to be deleted)	**single-click**
3	**Delete** button	**single-click**

Table 12.2-7. Delete a Resource - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
4	**OK** button	**single-click**

12.2.7 Shut Down Resource Definition Applications

When resource definition activities have been completed, the Message Handler, System Name Server, and Resource Model should be shut down to eliminate unneeded processes and allow other operators to gain access to the resource planning applications. If any of the three processes remains active, it is likely to interfere with subsequent attempts to launch resource planning applications.

Shutting down resource definition applications starts with the assumption that the **Resource Editor** GUI is currently being displayed.

Table 12.2-8 presents (in a condensed format) the steps required to shut down resource definition applications. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 To exit from the **Resource Editor** GUI when resource planning activities have been completed execute the following menu path:

File → Exit

- The **Resource Editor** GUI disappears.

2 After quitting the **Resource Editor** GUI **single-click** in the UNIX window used to start the resource definition applications.

3 Shut down the Message Handler, System Name Server, and Resource Model by entering:

EcPlRpSlayAll <MODE> <application_id>

- The **Message Handler** GUI disappears.

4 To obtain a list of active processes in the specified mode enter:

ps -ef | grep <MODE>

- A list of active processes in the specified mode is displayed.
- If an error message is received when **ps -ef | grep <MODE>** is entered, enter:

  ps -auxwww | grep <MODE>

5 Examine the list of processes running in the specified mode to determine whether the Message Handler, System Name Server, and Resource Model processes have actually been shut down.

- None of the following processes should be active:
  - EcPlRpRe
  - EcPlRpSi
  - EcPlRpTl
  - EcPlMsh
  - EcPlSns
  - EcPlRpRm

6 If any of the specified processes [especially the Message Handler, System Name Server, and/or Resource Model process(es)] is/are still active, terminate the active process(es)by entering:

kill -15 <process ID1> [<process ID2>] [<process ID3>] […]

7 Repeat Steps 4 through 6 as necessary.

Table 12.2-8. Shut Down Resource Definition Applications - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**File → Exit** (to quit the **Resource Editor** GUI)	**single-click**
2	**OK** button	**single-click**
3	UNIX window (Planning/Management Workstation)	**single-click**
4	**EcPlRpSlayAll <MODE>** <**application_id**>	**enter text, press Enter**
5	**ps -ef | grep <MODE>**	**enter text, press Enter**
6	Identify resource planning process(es) that has (have) not shut down	**read text**
7	**kill -15 <process ID1> [<process ID2>] [<process ID3>] […]** if necessary	**enter text, press Enter**

12.3 Scheduling Resources

The Resource Planner and Resource Manager are both involved in resource scheduling using the Resource Scheduler. The Production Planner and Production Monitor are involved in the implementation of ground events.

- Resource Planner processes resource reservation requests for ground events.
- Resource Manager commits resource reservations.

- Production Planner sends committed resource reservations (ground events) to Data Processing via the Planning Workbench.
- Production Monitor monitors execution of ground events in processing.

The following process is used for generating and implementing resource reservations (ground events):

- Personnel who have a need for Planning Subsystem or Data Processing Subsystem resources submit requests for time on specified resources to accomplish the non-routine activities that they plan to undertake.
  - Depending on DAAC policy, many personnel may have access to the resource planning applications for creating resource reservation requests.
  - Alternatively, personnel may have to contact the Resource Planner to have resource reservation requests entered for them.
- The Resource Planner reviews requests for resource reservations to determine if the requests are valid.
  - Request information includes a description of the activity, the resources required, the time period(s) for using the requested resource(s), comments explaining the variance from normal use.
  - Resource Planner may decide to forward the request to a “sponsor” for validation.
  - A sponsor is someone who evaluates a resource reservation request based on expertise that is particularly relevant to the resource reservation request.
- If the Resource Planner or sponsor determines that the request to reserve the resource is valid, the Resource Planner “approves” it along with all other requests that have been validated.
  - The set of all validated resource reservation requests is considered a draft Resource Plan.
- The scheduling software identifies conflicts (if any) in the draft Resource Plan and alerts the Resource Planner to the problem(s).
- If possible, the Resource Planner resolves all conflicts before presenting the proposed plan to the Resource Manager to have the resources committed.
  - However, during the conflict-resolution process the Resource Planner may have to consult with resource requesters and the Resource Manager to ensure that the reserved resources will not have adverse effects on the DAAC’s high-priority events.
- When the Resource Planner has achieved a conflict-free plan, it is presented to the Resource Manager to be implemented.
- The Resource Manager "commits" the resource plan, which signals the Planning Subsystem that the plan can be implemented.
  - Committing a plan actually involves committing all of the individual approved resource reservation requests that collectively make up the plan.

- All committed resource reservations are automatically included in the next production plan to be activated through the Planning Workbench and are subsequently sent to Data Processing.
  - Resource reservations/ground events are not sent to data processing until they have been included in a production plan.
  - Refer to Section 13, Production Planning, to see how production plans are created and activated.
- In Data Processing a ground event job for each resource reservation is sent to the specified resource(s) at the indicated start time.
  - If a data processing job is already using the specified resource(s) at the ground event's scheduled start time, the data processing job runs to completion before releasing the resource(s) to the ground event job.

Table 12.3-1, below, provides an Activity Checklist of Resource Scheduling activities.

Table 12.3-1. Resource Scheduling - Activity Checklist

Order	Role	Task	Section	Complete?
1	Resource Planner or DAAC Staff	Launch the Resource Scheduler	(P) 12.3.1	
2	Resource Planner or DAAC Staff	Create a Resource Reservation Request	(P) 12.3.2	
3	Resource Planner or DAAC Staff	Edit a Resource Reservation Request	(P) 12.3.3	
4	Resource Planner or Sponsor	Validate or Reject a Resource Reservation Request	(P) 12.3.4	
5	Resource Planner	Approve a Resource Reservation Request	(P) 12.3.5	
6	Resource Manager/Resource Planner	Commit Resource Reservation Requests	(P) 12.3.6	
7	Resource Planner	Review the Resource Timeline	(P) 12.3.7	
8	Resource Planner	Delete a Resource Reservation Request	(P) 12.3.8	
9	Resource Planner or DAAC Staff	Shut Down the Resource Scheduler	(P) 12.3.9	

12.3.1 Launch the Resource Scheduler

The **Resource Scheduler** is invoked from a UNIX command line prompt. Table 12.3-2 presents (in a condensed format) the steps required to launch the **Resource Scheduler**. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Access a terminal window logged in to the Planning/Management Workstation host.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 12.2.1).

2 In the terminal window, at the command line, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1 - Science Software Integration and Test (SSI&T)
  - TS2 - New Version Checkout
  - OPS - Normal Operations
- "utilities" is the directory containing the Planning Subsystem start-up scripts.

3 Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered.
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

4 Start the Resource Planning background processes by entering:

EcPlRpAllStart <MODE> <application_id>

- The Resource Planning background processes are launched.
- The **application_id** or **MSGSRV_ID** is a number from 1 to 5. It identifies the message service in use so messages can be directed to the proper message handler GUI. Consequently, it is a good idea to use the same application_id consistently during a resource planning session.

5 Start the **Resource Scheduler** GUI by entering:

EcPlRpSiStart <MODE> <application_id>

- The Resource Scheduler is launched.
- The **Resource Scheduler** Graphical User Interface (GUI) is accessed. The GUI displays the Resource Reservation List, activity type, and a series of buttons that enable the following operations:
  - **New...** Create a resource reservation request. (Section 12.3.2)
  - **Modify...** Edit or review the details of an existing resource reservation request. (Section 12.3.3)

– **Approve** Used to indicate that the resource reservation request(s) has (have) been validated and a draft resource plan can be created. Clicking on this button causes the Planning Subsystem to determine whether there are conflicts between this resource reservation and other reservations. The Planning Subsystem detects conflicts and reports them to the operator. (Section 12.3.5)
– **Commit globally** Commit all "approved" resource reservations; at this point the ground events will be accessible by the production planning software (Section 12.3.6); however, resource reservations/ground events are not sent to data processing until they have been included in an activated production plan.
– **Time Line** Display a timeline-oriented view of the resource plan. (Section 12.3.7)
– **Report** The **Report** option is disabled. The reports have been deleted from the system requirements.

Table 12.3-2. Launch the Resource Scheduler - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 12.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	Set the environment variables if necessary	**enter text, press Enter**
4	**EcPlRpAllStart <MODE>** <application_id>	**enter text, press Enter**
5	**EcPlRpSiStart <MODE>** <application_id>	**enter text, press Enter**

12.3.2 Create a Resource Reservation Request

Table 12.3-3 presents (in a condensed format) the steps required to create a Resource Reservation Request. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 If necessary, launch the **Resource Scheduler** GUI (refer to Section 12.3.1).
- The **Resource Scheduler** GUI is displayed.

2 From the **Resource Scheduler** GUI, **single-click** on the **New...** button to access the **Resource Reservation Request Edit/Definition** GUI.

3 Enter resource request identification information into the displayed fields. Press **Tab** to move from field to field. **NOTE:** Data that is system-generated is identified.
- **Request Name** Operator-provided name for the resource request. (required)
- **Edited Date** System-generated date of request entry.

- **Originator** Operator-provided name of the authorized user preparing the resource request.
- **Sponsor** Operator -provided name of the individual who is to review and validate the Resource Request (the subject-matter-expert). (required)

4 Enter resource scheduling information into the displayed fields. Press **Tab** to move from field to field.

- **Activity Type** Operator-provided description of the type of activity; selected by the operator from a selection list of valid options. (required)
- **Priority** Operator-provided priority for the activity. Use the slider to select the appropriate priority on a scale from 0 to 100. One (1) denotes the highest priority and 100 designates the lowest.
- **Description** Operator-provided description of the activity for which the resource is required. (required)
- **Resource...** See Section 12.3.2.1, below. (required)
- **Interval...** Not applicable to new resource reservation requests; may be applicable when editing a resource reservation request. See Section 12.3.3.1, below.

5 Enter duration information into the displayed fields to define the period over which the resource is required. Press **Tab** to move from field to field.

- **Start Date** Operator-provided start date of the resource request period. Enter in <**MM/DD/YYYY**> format. (required)
- **Start Time** Operator-provided start time of the resource request. Enter in <**hh:mm:ss**> format. (required)
  - **Start Time** must be later than the time when the resource reservation request will be saved; otherwise, it will not be possible to save the request.
- **Stop Date** Operator-provided stop date of the resource request period. Enter in <**MM/DD/YYYY**> format. If a reservation is to be repeated over some **frequency** (see below), the stop date specifies the end date in the date range of the reservation request. (required)
- **Stop Time** Operator-provided stop time of the resource request. Enter in <**hh:mm:ss**> format. (required)
- **Frequency** See Section 12.3.2.2, below.

6 Enter comments concerning the resource reservation request in the **Comment** field.

7 After data have been entered, **single-click** on the appropriate button from the following selections:

- **Save** to save data.
  - The resource reservation must be "saved" prior to validating or rejecting. After the request has been saved, it can then be **validated** or **rejected**.
  - The selections **Validated** and **Rejected** are further discussed in Section 12.3.3.
- **Clear** to clear entries. Once cleared, the entries are deleted from the system.
- **Cancel** to exit screen without saving the request.

Table 12.3-3. Create a Resource Reservation - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Scheduler** GUI (if necessary)	Use procedure in Section 12.3.1
2	**New…** button	**single-click**
3	**<resource identification information>**	**enter text, press tab**
4	**<resource scheduling information>**	**enter text, press tab**
5	**<duration information>**	**enter text, press tab**
6	**<comments>** (optional)	**enter text, press tab**
7	**Save** button	**single-click**

12.3.2.1 Selecting Resources...

Clicking on the **Resource…** button accesses a **Resources Selection** screen. The **Request Name** is blank and is to remain empty when creating a new resource reservation request. This screen provides a pair of lists: **Resources** and **Selected Resources**. The **Resources** list itemizes available resources. The **Selected Resources** list itemizes those resources that have been selected for incorporation into the resource reservation. The operator selects the desired resource(s) and, using the arrow buttons, moves the resource(s) from one list to the other list.

1 **Single-click** on your selections in the list and **single-click** on the desired arrow to move resources between the **Resources** and **Selected Resources** lists.

2 **Single-click** on one of the following buttons as appropriate:
- **OK** to save the selections and exit the screen.
- **Cancel** to exit the screen **without** saving changes.

12.3.2.2 Selecting Frequency

The **Frequency** option button provides the mechanism that allows the operator to specify whether the resource reservation request describes a one-time event or a recurring event. Clicking on **Frequency** allows the operator to specify options for periodic resource requests; that is, to specify the frequency of occurrence of a repeating resource need. Several options for expressing the frequency are available in the **Frequency** selection list box combined with a text field that provides a qualifier (i.e., number of days) for the **Every_?_days** selection only. The frequency specified defaults to **Once** to indicate that the resource need covers the entire time period covered by ‘Start Time’ and ‘Stop Time.’ Other options are identified in Table 12.3-4. The dates generated are inserted in the **Selected Intervals** list box, described in Section 12.3.2.3, below.

Table 12.3-4. Frequency Qualifiers

Frequency	Text Qualifier:	Result:
Once	--	The default. Resource reservation covering the period from the start time and stop time for the start date specified.
Daily	--	Resource reservation for every day, between the start date and end date, for the start time and end time specified.
Weekly	--	Resource reservation for every week occurring on the day specified by the start date, repeated until the end date as specified.
Every_2_weeks	--	Resource reservation occurring biweekly on the day specified by the start date, repeated until the end date as specified.
Monthly	--	Resource reservation for every month on the start day of the month, repeated until the end date as specified.
Mon_thru_Fri	--	Resource reservation for every Monday through Friday, between the start date and end date, for the start time and end time specified.
Mon_Wed_Fri	--	Resource reservation for every Monday, Wednesday, and Friday, between the start date and end date, for the start time and end time specified.
Tues_Thurs	--	Resource reservation for every Tuesday and Thursday, between the start date and end date, for the start time and end time specified.
Every_?_days	*n*	Resource reservation for every *n* days, between the start date and end date, for the start time and end time specified.
Weekend	--	Resource reservation for every Saturday and Sunday, between the start date and end date, for the start time and end time specified.

12.3.3 Edit a Resource Reservation Request

Table 12.3-5 presents (in a condensed format) the steps required to edit a Resource Reservation Request. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 If necessary, launch the **Resource Scheduler** GUI (refer to Section 12.3.1).
- The **Resource Scheduler** GUI is displayed.

2 From the **Resource Scheduler** GUI, **single-click** on the resource reservation request to be modified.

3 **Single-click** on the **Modify…** button to access the **Resource Reservation Request Edit/Definition** GUI.

4 Make the modifications to affected fields. (See Section 12.3.2, above.)

- **Interval...** is applicable when editing a resource reservation request if certain intervals are to be excluded from the resource reservation. See Section 12.3.3.1, below.

5 If appropriate at this time, **single-click** on either **Validated** or **Rejected**.

- **Validated** indicates that the reservation request is complete and 'makes sense'; that is, the request includes the appropriate resources consistent with the type of activity that is being proposed.
- **Rejected** indicates that the reservation request is rejected.
- At this time, the **Comment** field may also be updated.
- The **Status** field contains the status of the reservation request.
  - Status is system-generated based on operator-input in other fields.

6 After data is entered, **single-click** on the appropriate button(s):

- **Save** to save data and exit screen.
- **Clear** to clear entries. Once cleared, the entries are deleted from the system.
- **Cancel** to exit screen.

Table 12.3-5. Edit a Resource Reservation Request - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Scheduler** GUI (if necessary)	Use procedure in Section 12.3.1
2	**<resource reservation request>** (to be modified)	**single-click**
3	**Modify…** button	**single-click**
4	Make modifications to affected fields	Use procedure in Section 12.3.2
5	**Validated** button or **Rejected** button if applicable	**single-click**
6	**Save** button	**single-click**

12.3.3.1 Deselecting Intervals...

The **Interval…** button provides the mechanism to tailor a **Frequency-based** request by overriding selected intervals (**Note**: the initial resource reservation must be saved prior to tailoring frequency-based requests.). Selecting the **Interval…** button, displays a secondary screen that provides a pair of lists: **Unselected Intervals** and **Selected Intervals**. **Unselected Intervals** lists the dates that will not be reserved for the reservation request. **Selected Intervals** lists the dates that will be included for the request. The Selected Interval dates are automatically generated by the system, based upon the **Frequency** option selected (see Section 12.3.2.2, above). You can move them to or from the **Unselected Intervals** list to modify the automated list. Dates are moved from one list to the other by selecting the dates and using the arrow keys. The **Request Name** is also displayed.

1 **Single-click** on your selections and **single-click** on the desired arrow to move dates between the **Selected Intervals** and **Unselected Intervals** lists.

2 **Single-click** on one of the following buttons as appropriate:
- **OK** to save the selections and exit the screen.
- **Cancel** to exit the screen **without** saving changes.

12.3.4 Validate or Reject a Resource Reservation Request

All resource reservation requests must be validated and approved before scheduling. Validation is the process whereby a request is checked for completeness, and its purpose is deemed reasonable. After reviewing a resource reservation request, the Resource Planner may choose to consult with appropriate DAAC staff or assign a staff member ("sponsor") to validate a request. When the request is rejected, the status of the request is changed to "rejected" on the screen.

Table 12.3-6 presents (in a condensed format) the steps required to validate or reject a Resource Reservation Request. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 If necessary, launch the **Resource Scheduler** GUI (refer to Section 12.3.1).
- The **Resource Scheduler** GUI is displayed.

2 From the **Resource Scheduler** GUI, **single-click** on the resource reservation request to be modified.

3 **Single-click** on the **Modify…** button to access the **Resource Reservation Request Edit** GUI.

4 **Single-click** on either **Validated** or **Rejected**.
- **Validated** indicates that the reservation request is complete and 'makes sense'; that is, the request includes the appropriate resources consistent with the type of activity that is being proposed.
- **Rejected** indicates that the reservation request is rejected.
- At this time, the **Comment** field may also be updated.
- The **Status** field contains the status of the reservation request.
  - Status is system-generated based on operator-input in other fields.

5 After data is entered, **single-click** on the appropriate button(s):
- **Save** to save data.

- **Clear** to clear entries. Once cleared, the entries are deleted from the system.
- **Cancel** to exit screen.

Table 12.3-6. Validate or Reject a Resource Reservation Request - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Scheduler** GUI (if necessary)	Use procedure in Section 12.3.1
2	**<resource reservation request>** (to be modified)	**single-click**
3	**Modify…** button	**single-click**
4	**Validated** button or **Rejected** button as appropriate	**single-click**
5	**Save** button	**single-click**

12.3.5 Approve a Resource Reservation Request

The **Approve** button is used when all reviews that are a part of the resource planning process have taken place and there are no objections to the resource usage as described by the request. Clicking on this button will verify that there are no conflicts between this resource reservation and other reservations. If conflicts are detected, a screen will pop up listing the conflicts to be addressed for resolution. Click **OK** to collapse the pop-up screen. Clicking on **Approve** generates the pop-up screen again (if conflicts exist). Approval occurs after a request has been validated and the event time is acceptable.

Table 12.3-7 presents (in a condensed format) the steps required to approve a Resource Reservation Request. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 If necessary, launch the **Resource Scheduler** GUI (refer to Section 12.3.1).
- The **Resource Scheduler** GUI is displayed.

2 From the **Resource Scheduler** GUI, **single-click** on the resource reservation request to be approved.

3 **Single-click** on the **Approve** button.
- If there are resource conflicts resulting from the attempt to approve the resource reservation request, a pop-up dialogue box appears indicating that the approval failed and making reference to the **Message Handler** GUI for further information.

4 **Single-click** on the **OK** button to collapse the pop-up dialogue box.

- If there are no resource conflicts to be resolved, the entry in the Status column of the **Resource Scheduler** GUI indicates that the request is "Approved" (changes from "Validated"). [End of procedure.]
- If there are resource conflicts to be resolved, the entry in the Status column of the **Resource Scheduler** GUI indicates that the request has "Conflicts" (changes from "Validated"). [Continue with Step 5.]

5 If there are resource conflicts to be resolved, examine the information displayed on the **Resource Scheduler** GUI.

- Although the pop-up dialogue box makes reference to the **Message Handler** GUI for further information, no relevant data seems to be displayed there. Therefore, it is more appropriate to check for conflicts in the duration and frequency information for the resource reservation requests displayed on the **Resource Scheduler** GUI. When more than one resource reservation request is scheduled for the same date and time, there may be a conflict (if the same resource is specified in the requests).
- It may be necessary to examine individual resource reservation requests in detail. If so, use the procedure to **Edit a Resource Reservation Request** (Section 12.3.3).

6 If necessary, consult with the resource requester(s), Resource Manager and other personnel to determine which resource reservation request(s) to modify or delete in order to create a conflict-free resource plan.

7 If applicable, go to the procedure to **Delete a Resource Reservation Request** (Section 12.3.8) and delete resource reservation request(s) as necessary to resolve the conflicts.

8 If applicable, go to the procedure to **Edit a Resource Reservation Request** (Section 12.3.3) and modify/validate resource reservation request(s) as necessary to resolve the conflicts.

9 If applicable, return to Step 2 to approve a modified resource reservation request.

- The modified procedure must have been "validated." If necessary, refer to the procedure to **Validate or Reject a Resource Reservation Request** (Section 12.3.4).

Table 12.3-7. Approve a Resource Reservation Request - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Scheduler** GUI (if necessary)	Use procedure in Section 12.3.1
2	**<resource reservation request>** (to be approved)	**single-click**

Table 12.3-7. Approve a Resource Reservation Request - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
3	**Approve** button	**single-click**
4	**OK** button	**single-click**
5	If there are resource conflicts to be resolved, examine the information displayed on the **Resource Scheduler** GUI	**read text**
6	Resolve conflicts as necessary	Use procedures in Sections 12.3.8, 12.3.3, and/or 12.3.4

12.3.6 Commit Resource Reservation Requests

Clicking on the **Commit globally** button commits all approved reservation requests and makes them accessible to Production Planning. All committed resource reservations are automatically included in the next production plan to be activated through the Planning Workbench and are subsequently sent to Data Processing. Note that resource reservations/ground events cannot take effect until they have been sent to Data Processing as part of an activated production plan. (Refer to Section 13, Production Planning, to see how production plans are created and activated.)

In Data Processing a "ground event" job for each resource reservation is sent to the specified resource(s) at the indicated start time. If a data processing job is already using the specified resource(s) at the ground event's scheduled start time, the data processing job runs to completion before releasing the resource(s) to the ground event job.

Table 12.3-8 presents (in a condensed format) the steps required to commit a Resource Reservation Request. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 From the **Resource Scheduler** GUI, **single-click** on the **Commit globally** button.

- Status shows **Committed** for all previously **Approved** requests.

2 To view a graphical representation of the resource plan execute the procedure in Section 12.3.7.

3 To exit from the **Resource Scheduler** GUI execute the procedure in Section 12.3.9.

NOTE: Resource reservations/ground events are not sent to data processing and cannot be implemented until they have been included in a production plan. Refer to Section 13, Production Planning, for the procedure on creating (including activating) production plans.

Table 12.3-8. Commit Resource Reservation Requests - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**Commit globally** button	**single-click**
2	View a graphical representation of the resource plan if desired	Use the procedure in Section 12.3.7
3	Exit from the **Resource Scheduler** GUI if desired	Use the procedure in Section 12.3.9

12.3.7 Review the Resource Timeline

The Resource Planning utilities allow the operator to view the Resource Plan as a timeline. Table 12.3-9 presents (in a condensed format) the steps required to review the Resource Timeline. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 If necessary, launch the **Resource Scheduler** GUI (refer to Section 12.3.1).
- The **Resource Scheduler** GUI is displayed.

2 From the **Resource Scheduler** GUI **single-click** on the **Timeline** button.
- The **Resource Timeline** GUI is displayed.
- The display represents a set of resources, arranged along the left side of the screen and some period of time as indicated across the top edge of the screen.
- The use of a resource over a period of time is represented by one or more 'resource reservation' bars across the screen.
- A bar represents a time period during which a resource reservation has been planned for the resource.
  - Each bar has the name of the resource reservation and a brief description.
  - For time periods during which a reservation has not been placed against a resource, that resource is planned for use by a default activity, e.g., science processing computers will be used for science processing unless a reservation has been placed against that resource.
  - Scroll bars allow scrolling up and down through the full list of resources and left and right in time.

3 Adjust the **Resource Timeline** window size and the view of the timeline as necessary using the mouse.
- Grab a corner of the timeline window with the cursor and resize the window as desired.
- Scroll up or down through the full list of resources.
- Scroll left or right to go backward or forward in time.

4 If a different time scale (start and end dates and times) is desired, perform Steps 5 through 7; otherwise, go to Step 8.

5 Execute the following menu path:

Display → Change Time Scale

- The **plan window edit** window is displayed.

6 In the **Plan Win Start** and **Plan Win End** fields of the **plan window edit** window enter the date and time for the desired start and end times using the following format:

<DD MMM YYYY hh:mm:ss>

7 When the appropriate date and time have been entered, **single-click** on the appropriate button from the following selections:

- **OK** - to accept the changes and dismiss the **plan window edit** window.
- **Apply** - to accept the changes without dismissing the **plan window edit** window.
- **Cancel** - to cancel the changes and dismiss the **plan window edit** window.

8 If a different time span is desired, **single-click** and **hold** on the **Show** option button, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.

- Options are: **1 hr, 4 hr, 8 hr, 12 hr, 24 hr, 48 hr, 4 day, 1 week, 2 week, 1 month, full scale.**

9 If no resources are displayed on the GUI or if different resources should be displayed, perform Steps 10 through 14; otherwise, go to Step 15.

10 Execute the following menu path:

Display → Change resources

- The **Resource edit** window is displayed.

11 If adding resource(s) from the **Available Resources** list to the **Viewed Resources** list, select (highlight) the resource(s) to be added, then click on the **Add** button to move the resource(s) to the **Viewed Resources** list.

- Highlighted resource(s) appear(s) on the **Viewed Resources** list.

12 If deleting resource(s) from the **Viewed Resources** list, select (highlight) the resource(s) to be removed, then click on the **Del** button to remove the resource(s) from the **Viewed Resources** list.

- Highlighted resource(s) disappear(s) from the **Viewed Resources** list.

13 If changing the order in which resources are listed in the **Viewed Resources** list, select (highlight) the resource to be moved, then **single-click** on the up or down arrow as necessary to reposition the selected resource.

- Highlighted resource changes position in the **Viewed Resources** list.

14 When the **Viewed Resources** list contains the desired set of resources, **single-click** on the appropriate button from the following selections:

- **OK** - to accept the changes and dismiss the **Resource edit** window.
- **Apply** - to accept the changes without dismissing the **Resource edit** window.
- **Cancel** - to cancel the changes and dismiss the **Resource edit** window.

15 If different color coding of the timeline is desired, perform Steps 16 through 20; otherwise, go to Step 21.

16 Execute the following menu path:

Display → Change colors

- The **Color Selections** window is displayed.

17 **Single-click** on the name of one of the resource reservations to be recolored.

- The resource reservation is highlighted.

18 **Single-click** on the desired color (in the color palette) to be applied to the highlighted resource reservation.

19 Repeat Steps 17 and 18 as necessary.

20 When the appropriate color changes have been made, **single-click** on the appropriate button from the following selections:

- **OK** - to accept the changes and dismiss the **Color Selections** window.
- **Apply** - to accept the changes without dismissing the **Color Selections** window.
- **Cancel** - to cancel the changes and dismiss the **Color Selections** window.

21 Observe the resource reservation information displayed on the **Resource Timeline** GUI.

22 Repeat the previous steps as necessary.

23 If it becomes necessary to exit from the timeline GUI execute the following menu path: **File → Quit**

Table 12.3-9. Review the Resource Timeline - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Scheduler** GUI (if necessary)	Use procedure in Section 12.3.1
2	**Timeline** button	**single-click**
3	**Display → Change Time Scale**	**single-click**
4	**<plan window start date and time>**	**enter text**
5	**<plan window end date and time>**	**enter text**
6	**Ok** button	**single-click**
7	**<time span>**	**single-click**
8	**Display → Change Resources**	**single-click**
9	**<resources>** (to be viewed)	**single-click**
10	**Add** button	**single-click**
11	**<viewed resource>** (to be reordered)	**single-click**
12	**<up>** arrow or **<down>** arrow (as necessary to reorder viewed resources)	**single-click**
13	**Ok** button	**single-click**
14	**Display → Change Colors**	**single-click**
15	**<resource reservation>** (to be recolored)	**single-click**
16	**<color>** for resource reservation	**single-click**
17	**Ok** button	**single-click**
18	Observe the resource reservation information	**read text**
19	**File → Quit** (to quit the timeline)	**single-click**

12.3.8 Delete a Resource Reservation Request

Table12.3-10 presents (in a condensed format) the steps required to delete a resource reservation request. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 If necessary, launch the **Resource Scheduler** GUI (refer to Section 12.3.1).
- The **Resource Scheduler** GUI is displayed.

2 From the **Resource Scheduler** GUI, highlight (click on) the resource reservation request you want to delete.

3 Execute the following menu path:

File → Delete

- Status shows "Deleted" for the selected request. The resource reservation request is not removed from the database at this point and is available for future reporting but will have no impact on resource planning. Resource reservations are removed from the Resource reservations (PDPS) database through routine database maintenance activities.

4 To exit from the **Resource Scheduler** GUI execute the procedure in Section 12.3.9.

Table 12.3-10. Delete a Resource Reservation Request - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch the **Resource Scheduler** GUI (if necessary)	Use procedure in Section 12.3.1
2	**<resource reservation request>** (to be deleted)	**single-click**
3	**File → Delete**	**single-click**
4	Exit from the **Resource Scheduler** GUI if desired	Use procedure in Section 12.3.9

12.3.9 Shut Down the Resource Scheduler

When resource scheduling activities have been completed, the Message Handler, System Name Server, and Resource Model should be shut down to eliminate unneeded processes and allow other operators to gain access to the resource planning applications. If any of the three processes remains active, it is likely to interfere with subsequent attempts to launch resource planning applications.

Shutting down the Resource Scheduler starts with the assumption that the **Resource Scheduler** GUI has been launched and is currently being displayed.

Table 12.3-11 presents (in a condensed format) the steps required to shut down resource scheduling applications. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 To exit from the **Resource Scheduler** GUI when resource planning activities have been completed execute the following menu path:

File → Exit

- The **Resource Scheduler** GUI disappears unless there are resource reservation requests with a status of "approved".
- If there are any resource reservation requests with a status of "approved" listed on the **Resource Scheduler** GUI, a **Close Application** pop-up dialogue box is displayed with a message "Status of the listed reservations" and a list of the resource reservation requests with "approved" status.

2 If the **Close Application** pop-up dialogue box is displayed, **single-click** on the appropriate button from the following selections:

- **Ok** - to quit the **Resource Scheduler** GUI and dismiss the dialogue box.
  - Selecting **Ok** effectively commits all "approved" Resource Reservations.
- **Cancel** - to dismiss the dialogue box and return to the **Resource Scheduler** GUI.

3 After quitting the **Resource Scheduler** GUI **single-click** in the UNIX window used to start the resource scheduling applications.

4 Shut down the Message Handler, System Name Server, and Resource Model by entering:

EcPlRpSlayAll <MODE> <application_id>

- The **Message Handler** GUI disappears.

5 To obtain a list of active processes in the specified mode enter:

ps -ef | grep <MODE>

- A list of active processes in the specified mode is displayed.
- If an error message is received when **ps -ef | grep <MODE>** is entered, enter:

  ps -auxwww | grep <MODE>

6 Examine the list of processes running in the specified mode to determine whether the Message Handler, System Name Server, and Resource Model processes have actually been shut down.

- None of the following processes should be active:
  - EcPlRpRe
  - EcPlRpSi
  - EcPlRpTl
  - EcPlMsh
  - EcPlSns
  - EcPlRpRm

7 If any of the specified processes [especially the Message Handler, System Name Server, and/or Resource Model process(es)] is/are still active, terminate the active process(es)by entering:

kill -15 <process ID1> [<process ID2>] [<process ID3>] […]

8 Repeat Steps 5 through 7 as necessary.

Table 12.3-11. Shut Down the Resource Scheduler - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**File** → **Exit** (to quit the **Resource Scheduler** GUI)	**single-click**
2	**OK** button	**single-click**
3	UNIX window (Planning/Management Workstation)	**single-click**
4	**EcPlRpSlayAll <MODE>** <**application_id**>	**enter text, press Enter**
5	**ps -ef | grep <MODE>**	**enter text, press Enter**
6	Identify the resource scheduling process(es) that has (have) not shut down	**read text**
7	**kill -15 <process ID1> [<process ID2>] [<process ID3>] […]** if necessary	**enter text, press Enter**

12.4 Tuning System Parameters

The values assigned to system parameters affect the functioning and performance of the system. When certain parameters are modified, the system operates differently. Changes to some other parameters may not appear to affect the system although there may in fact be subtle effects. In any case before system parameters are modified it is essential to understand what will happen to system functioning and performance.

Many system parameters may be subject to control by Configuration Management (CM). When making or requesting a change to system parameters, the CM process at the particular site must be followed (if applicable).

Values are assigned to Data Processing Subsystem and Planning Subsystem parameters in the following databases:

- PDPS database.
- Configuration Registry database.

The Configuration Registry Server provides a single interface (via a Sybase server) for retrieving configuration attribute-value pairs for ECS servers from the Configuration Registry database.

When ECS servers are started, they access the Configuration Registry Database to obtain needed configuration parameters.

The Database Administrator has access to a Configuration Registry GUI for viewing and editing configuration data in the database. Therefore, it is necessary to coordinate with the Database Administrator when changes to configuration parameters are needed. Also, as previously mentioned, changes to configuration-controlled parameters are subject to approval through the site CM process.

Default and adjusted values assigned to system parameters vary from site to site. For guidance concerning the assignment of values to parameters included in the Configuration Registry refer to document 910-TDA-022, *Custom Code Configuration Parameters for ECS*. The document is available at http://cmdm.east.hitc.com/baseline/ under "Technical Documents."

The following parameters are examples of parameters whose values may be modified to enhance system functioning or performance:

- AppLogSize [parameter applies to all servers].
  - Maximum size of the application log (ALOG) file for a particular application.
  - Recommended size varies considerably depending the nature of the application for which the file is being written.
- AppLogLevel [parameter applies to all servers].
  - Level of detail provided in the ALOG file for a particular application.
  - Acceptable values are 0, 1, 2, or 3.
  - A setting of "0" provides the most data.
- DebugLevel [parameter applies to all servers].
  - Level of detail provided in the debug log file for a particular application.
  - Normally acceptable values are 0, 1, 2, or 3.
  - A setting of "0" turns off logging; a setting of "3" provides a significant amount of data.
- DpPr_MAX_RETRIES [EcDpPrEM and EcDpPrDeletion parameter (also EcDpPrQaMonitorGUI and several Science Software Integration and Test programs)].
  - Number of retries (e.g., 30) to the Science Data Server for acquires/inserts before giving up.
- DpPr_WAIT_PERIOD [EcDpPrEM and EcDpPrDeletion parameter (also EcDpPrQaMonitorGUI and several Science Software Integration and Test programs)].
  - Time in seconds (e.g., 120) to wait between retries to the Science Data Server.
- DpPrRM_MAX_RETRIES [EcDpPrEM, EcDpPrGE, EcDpPrJobMgmt, EcDpPrDeletion parameter].
  - Maximum number (e.g., 100) of attempts to allocate a computer resource.
- DpPrRM_RETRY_PERIOD [EcDpPrEM, EcDpPrGE, EcDpPrJobMgmt, EcDpPrDeletion parameter].
  - Number of seconds (e.g., 120) between retries when trying to allocate a resource.

- DpPrMaxConcurrentDPRs [EcDpPrJobMgmt parameter].
  - Maximum allowed jobs.
  - Three integer values (e.g., 100 100 100) are assigned to DpPrMaxConcurrentDPRs; the first for routine processing; the second for on-demand processing; and the third for reprocessing jobs.
- DpPrMinConcurrentDPRs [EcDpPrJobMgmt parameter].
  - Minimum allowed jobs.
  - Three integer values (e.g., 0 0 0) are assigned to DpPrMaxConcurrentDPRs; the first for routine processing; the second for on-demand processing; and the third for reprocessing jobs.
  - Minimum number of concurrent DPRs for each job class (i.e., routine, on demand, reprocessing) NOT CURRENTLY USED.
- DpPrAutoSysMaxDPRs [EcDpPrJobMgmt parameter].
  - Total number of jobs (e.g., 100) allowed in AutoSys.
- DpPrDeleteFailedPGEJobs [EcDpPrJobMgmt parameter].
  - If TRUE, failed PGE Jobs are removed by Job Management, as necessary, when space is needed for another job that is ready to run. This is recommended to keep job management straightforward. However, this may be confusing for the operator, since they may not get a chance to see the failure if the system is busy.
  - If FALSE (the usual value), failed PGE Jobs are left in AutoSys. They must not be removed manually from AutoSys, however, since they will be removed by the Production Request Editor when a Production Request or DPR is cancelled.
- DBConnections [EcPoConnections (includes EcPlSubMgr, EcPlOdMgr, EcDpPrDeletion, EcDpPrJobMgmt and EcDpPrJobMgmtClient) parameter].
  - Number of connections needed by a particular application (e.g., 10 for EcPlOdMgr).
  - Optional parameter that specifies the number of connections to maintain in the connection pool.
  - The parameter is a list of positive integers. There must be one entry for each DbHandle in the DbHandleList.
  - Generally it should be set to the maximum number of connections that are expected to be used simultaneously in a process. If one connection per thread is used, this will be the same as the number of concurrent threads expected to execute. When the pool is used up there is a performance penalty to allocate and deallocate connections on the fly.
  - If this parameter is not specified or is given as “NONE”, it defaults to 1.
- SleepDelayForFailures [EcPlSubMgr parameter].
  - Amount of time in seconds (e.g., 60) to wait before reprocessing failed notifications. If the specified value is less than 60, a default value of 60 seconds would be assumed.
  - Duration of the sleep delay used by the failed notification thread in seconds.
  - Less frequent checking can increase speed for the other threads.

- SleepDelayForTimers [EcPlSubMgr parameter].
  - Amount of time in seconds (e.g., 60) the Subscription Manager should sleep between checking for expired timers. It should be set to the minimum amount of time a timer will be set for at this DAAC. The minimum it can be set to is 60 seconds.
  - Duration of sleep delay used by the timer checking thread in seconds.
  - Less frequent checking can increase speed for the other threads.
- SleepDelayForExp [EcPlOdMgr parameter].
  - Sleep delay for expiration thread in seconds (e.g., 86400).
  - Should be considerably greater than the sleep delay for completion threads (SleepDelayForCmp).
- SleepDelayForCmp [EcPlOdMgr parameter].
  - Sleep delay for completion threads in seconds (e.g., 300).
  - Should be considerably less than the sleep delay for expiration threads (SleepDelayForExp).
- SocketLimit [EcDpPrDeletion, EcDpPrJobMgmt, EcPlOdMgr, EcPlSubMgr parameter].
  - Number of connections (e.g., 200) to a server through the Hubble Space Telescope (HST) sockets middleware.
  - Too low a number misses connections.
  - Too high a number may adversely affect the memory of the server's host.

NOTE: When the value assigned to a parameter has been changed and saved in the Configuration Registry, the modified value does not take effect until the affected server has been restarted. For example, if the debug level for the Subscription Manager log has been changed from “2” to “3” in the Configuration Registry, the modification does not affect the recording of data in the log until after a warm restart of the Subscription Manager (at which time the server would read the parameters in the Configuration Registry).

Table 12.4-1, below, provides an Activity Checklist table of System Tuning activities.

Table 12.4-1. Tuning System Parameters - Activity Checklist

Order	Role	Task	Section	Complete?
1	Resource Planner/ Production Planner/ Production Monitor	Monitor the Load on Processing Resources	(P) 12.4.1	

12.4.1 Monitor the Load on Processing Resources

The Production Planner and Production Monitor should work with the Resource Planner to make optimum use of processing resources. The Resource Planner allocates the disk partitions, CPUs, and RAM available for processing among the active modes (e.g., OPS, TS1, TS2). The Production Planner and Production Monitor monitor the load on the processing resources.

The Resource Planner assigns the bulk (typically 60% - 80%) of the processing resources to the OPS mode. The remainder of the processing assets are divided among the modes used for SSI&T and new version software checkout.

The Production Planner and Production Monitor monitor the load on the processing resources to identify whether the actual load is appropriately distributed among modes. They inform the Resource Planner of under- or over-use of resources as allocated.

When monitoring the load on the processing resources, the Production Planner and Production Monitor should take the following considerations into account:

- Disk space allocated to OPS mode is likely to be used to capacity.
- Disk space assigned to the other two modes may not fill up.
- There is no one-to-one mapping of CPU allocation with actual CPUs on the science processor.
- The operating system (OS) takes care of true CPU and RAM allocation.
  - Actual CPU usage during processing is limited by the OS.
  - If ten CPUs have been specified for a particular mode, only ten Data Processing Requests (DPRs) can be running the Execute job at a given time.
  - What is really being defined is the maximum number of DPRs that will execute at a given time.
- CPUs can be over-allocated or under-allocated as necessary to get the most out of the CPUs on each science processor.
- If monitoring indicates that the processor is underused when OPS mode is at full processing capacity, the number of CPUs allocated to OPS mode could probably be increased.
- If the science processor is at full capacity when OPS mode is at full processing capacity (and the processor may be overworked) the number of CPUs allocated to OPS mode should be reduced.
- Random-access memory (RAM) is subject to the same considerations as CPUs.
  - RAM can be over-allocated or under-allocated as necessary to get the most out of the memory on each science processor.

12.4.2 Strategies for Tuning

A scenario that demonstrates how DPRs might be processed under a particular set of conditions and some strategies for tuning the system are presented in the paragraphs that follow. The processing conditions include the following types of items:

- The total number of jobs allowed into AutoSys.
- The number of CPUs available for processing.
- Characteristics of the PGEs to be processed.

The total number of jobs allowed into AutoSys is controlled by the DpPrPgeLimits table in the PDPS database. An example of some of the types of data maintained in the DpPrPgeLimits table is shown in Table 12.4-2.

Table 12.4-2. Example of PDPS Database DpPrPgeLimits Table Contents (Selected Columns)

computerName [Virtual Computer]	pgeId	maxConcurrent [DPRs]
A	1	20
B	1	20
A	2	20
B	2	20

The scenario assumes that each of the virtual computers (i.e., A and B) listed in Table 12.4-2 has 16 CPUs. (There are 32 CPUs total.)

Relevant PGE characteristics are shown in Table 12.4-3.

Table 12.4-3. PGE Characteristics

PGE	# CPUs Used	Average Execution Time	Average Stage Time	Destage Time
1	1	5 minutes	5 minutes	5 minutes
2	1	60 minutes	5 minutes	5 minutes

Assuming that 100 DPRs of each type (i.e., PGE 1 and PGE 2 - 200 DPRs total) are ready to run and are released at once into AutoSys, the following actions occur:

- Eighty (80) DPRs enter AutoSys. The remaining 120 are queued, with their assignments already made:
  - Machine (Virtual Computer) A: 20 PGE 1s start staging; 30 PGE 1s are queued on Machine A; 20 PGE 2s start staging; 30 PGE 2s are queued on Machine A.

- Machine (Virtual Computer) B: 20 PGE 1s start staging; 30 PGE 1s are queued on Machine B; 20 PGE 2s start staging; 30 PGE 2s are queued on Machine B.

- After about five (5) minutes, all 80 DPRs that were staging have finished staging and are ready for execution. However, only 32 CPUs are available.
- The first 32 DPRs that ask for CPUs get them and start running [sixteen (16) on Machine A and sixteen (16) on Machine B]. Forty-eight (48) DPRs are waiting.
  - Assuming that in the Registry database DpPrRM_RETRY_PERIOD is set to 120 seconds and DpPrRM_MAX_RETRIES is set to 100, the waiting DPRs keep trying every two minutes for up to 100 times each before timing out (after 200 minutes).
  - Note that in this example timing out is a real possibility.
- The quick jobs complete processing after five (5) minutes, freeing up sixteen (16) CPUs. In the current example, the sixteen (16) CPUs are subsequently occupied with about eight (8) five-minute PGEs and eight (8) 60-minute PGEs because CPUs are given randomly to whichever DPR gets back first to asking for them after waiting for the retry period (i.e., 120 seconds). Priorities are not used.
  - At first, there was a 50:50 ratio of fast:slow DPRs, now there is a 25:75 ratio of fast:slow. After another five (5) minutes, the ratio becomes 12.5:87.5 fast:slow, so 87.5 % of the CPUs are occupied by 60-minute DPRs.
- Apparently, the 60-minute DPRs tend to dominate the CPUs. After one (1) hour the first batch of sixteen (16) 60-minute PGEs vacates the CPUs to be replaced by eight (8) five-minute PGEs and eight (8) 60-minute PGEs, but the five-minute PGEs become extinguished again by the slow ones.
  - If the staging and destaging times were not the same (so the DPRs didn't have the same opportunity to hit the execution stage at the same time) the scenario would proceed differently.

Various strategies can be employed to tune the system:

- Limit the number of DPRs through the use of the DpPrPgeLimitsTable.
  - In the preceding example if the number of slow DPRs allowed into AutoSys is less than the number of CPUs, there is always a channel for the fast jobs to squeeze through.
  - The big disadvantage to this approach is that the slow jobs are also being prevented from staging.
- Increase the declared number of CPUs for the processors to more than the actual number (overallocate CPUs).
  - This approach allows more of each type of PGE into the science processors.
  - The disadvantage is that it could overwhelm the science computers. However, they are kept busy.
- Create new virtual computers (assigning CPUs on the processors to them) and assign (via the DpPrPgeLimits table) PGEs to run on the new virtual computers.
  - This approach is another way to guarantee bandwidth (CPUs) to PGEs.

- The disadvantage of this approach is that some CPUs could remain idle, not being seen by one of the virtual computers.
- In the past, there may have also been some code problems with supporting this, but those difficulties should have been resolved.

Probably some combination of the first two of the preceding strategies is best; i.e., increase the number of declared CPUs to be more than the total number of slow jobs allowed into AutoSys, always leaving some CPUs for a channel of fast jobs. The total number of faster-moving jobs should be increased to make sure that there is always be a queue of them available to get their channel occupied.

The staging and destaging times have to be accounted for and this could change things in terms of using the DpPrPgeLimits table and the number of CPUs per processor to tune the job flow.

Also, it is important to perform regular garbage collection on all of the virtual computers. Procedures for cleaning the PDPS database and DPS disks (i.e., "garbage collection") are provided in Chapter 13, Production Planning.

12.5 Troubleshooting Resource Planning Problems

Troubleshooting is a process of identifying the source of problems on the basis of observed trouble symptoms. One common source of problems involves the reliance on messages or data from other subsystems. However, unlike many other operational areas in ECS, Resource Planning does not have interfaces with many other subsystems. Consequently, problems with Resource Planning can usually be traced to either some part of the Planning Subsystem or the ECS infrastructure.

Table 12.5-1, below, provides an Activity Checklist for troubleshooting Resource Planning problems.

Table 12.5-1. Troubleshooting Resource Planning Problems - Activity Checklist

Order	Role	Task	Section	Complete?
1	Resource Planner	Troubleshoot a Resource Planning Problem	(P) 12.5.1	
2	Resource Planner	Check Log Files	(P) 12.5.2	
3	Resource Planner	Check Database Connections	(P) 12.5.3	

Fault Recovery

Each request that crosses a client/server boundary is assigned a system-unique identifier referred to as an RPC ID. (RPC refers to Remote Procedure Call, the mechanism by which requests are submitted from client to server.) The RPC ID facilitates the automatic fault recovery events that occur whenever there is a client or server failure.

- As a request propagates through the system, each associated client/server exchange is assigned a unique RPC ID.
  - The RPC ID for each interaction is derived from the previous RPC ID received by the client for the request. Consequently, all RPC IDs associated with a given request have a common portion that relates the various client/server calls to one another.
  - Given the previous RPC ID, clients consistently reproduce the same RPC ID that was submitted to the server on the subsequent event.
- The concept of reproducible RPC IDs is central to the ECS fault recovery capability.
  - When requests are retried from client to server, they are always submitted with the same RPC ID that was used in the original submission of the request, even if either the client or server has crashed between retries.
- The RPC ID is also central to the check-pointing aspect of fault recovery.
  - As requests arrive at fault recovery-enabled servers, they are recorded in a persistent store (typically a database), tagged with the RPC ID, which identifies the request.
  - As the request is serviced, check-pointing state information may be updated in the persistent store, up to and including the completion status of the request.
  - This allows the servers to resume servicing from the last check-pointed state, particularly upon resubmission from a client.

Fault Handling

Failure events are classified according to the following three severity levels:

- Fatal error.
  - Returned when a request cannot be serviced, even with operator intervention.
  - For example, if a request is made to distribute data via ftp to a non-existent host, the request is failed with a fatal error.
- Retry error.
  - Potentially recoverable error.
  - Normally, a retry error would be returned to the client only when the server cannot recover from the error automatically.
  - A retry error may require operator assistance during recovery. For example, a tape left in a tape drive might have to be removed manually.
- Warning.
  - Provided when operations can proceed without interruption, but an unexpected circumstance was detected.
    - For example, when using the Resource Scheduler GUI, the Resource Planner would enter a new name for a resource reservation request after being notified that there was a previously existing resource reservation request with the name that had been entered.

Transient errors (such as network errors) are always retry errors.

- In general, clients and servers that experience transient retry errors first attempt to recover by retrying the operation automatically.
- One special case of this is "rebinding," which refers to the process by which a client automatically attempts to re-establish communication with a server in the event communication is disrupted.
  - The disruption may be caused by transient network failure, or by the server crashing or being brought down.
  - In any case, the client automatically attempts to reconnect to the server for a configurable period of time on a client-by-client basis.

ECS processes encountering an error or receiving an error from a server request can either pass the error back to a higher-level client or present it to the operator for operator intervention.

Client Crash and Restart

In general when a client crashes, the server continues to service the requests that were in process at the time of the client's crash. When a client restarts in the ECS system, it sends a restart notification to each server with which it interacts.

- Clients notify servers that they have come up either "cold" or "warm."
- Generally, the notification temperature sent to the server matches the temperature at which the client process is restarted.

The default server behavior in response to startup notification from a client is as follows:

- Warm Notification.
  - Outstanding requests for the restarted clients remain available in the persistent store.
  - The outstanding requests may be resubmitted by the client, and are serviced to completion upon resubmission.
  - Associated resources are left allocated until the requests are completed.
- Cold Notification.
  - All outstanding requests for the restarted client are cancelled.
  - If the client resubmits any cancelled request using the same RPC ID (e.g., by pressing the Retry button from an operator GUI), it is failed with a fatal error due to the client cold startup notification.
  - Any resources associated with the cancelled requests are released and reclaimed by the system.

Server Crash and Restart

When a server crashes, clients cannot continue to submit requests for processing.

- Synchronous requests in progress result in a Distributed Computing Environment (DCE) exception being thrown back to the client process, which enters a rebinding failure recovery mode (as previously mentioned).

- Attempts to submit requests while the server is down result in the client blocking until a communication timeout has been reached.
- Although DCE has been replaced by socket-based library calls (i.e., CCS Middleware), the DCE exception code is handled by the CCS Middleware.

When a server restarts, it may perform various resynchronization activities in order to recover from an unexpected termination.

- In the event of a server cold start or cold restart, the server typically cancels all outstanding requests and reclaims all associated resources.
- In general, existing request queues are retained for warm restarts and cleared for cold starts or cold restarts.

12.5.1 Troubleshoot a Resource Planning Problem

1 If it is not possible to log in to the Planning Subsystem host, ask the Operations Controller/System Administrator to verify that the host is "up."

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**

2 If the GUI (i.e., the **Resource Editor** GUI or the **Resource Scheduler**) is not displayed when the start-up script has been properly invoked, ensure that the DISPLAY variable was set properly.

- For detailed instructions refer to the applicable procedure.
  - **Launch the Resource Editor** (Section 12.2.2).
  - **Launch the Resource Scheduler** (Section 12.3.1).

3 If an error message is received indicating that SNS (System Name Server) and/or Resource Model is/are in use using the selected Application ID and if working in a different mode from the person using the selected Application ID, use a different Application ID.

- For detailed instructions refer to the applicable procedure.
  - **Launch the Resource Editor** (Section 12.2.2).
  - **Launch the Resource Scheduler** (Section 12.3.1).

4 If an error message is received indicating that SNS (System Name Server) and/or Resource Model is/are in use using the selected Application ID and if working in the same mode as the person using the selected Application ID, coordinate use of Planning applications with the other user and/or the System Administrator.

5 If an error message associated with the Resource Editor is received, refer to Table 12.5-2, Resource Editor User Messages.

- The table is adapted from the corresponding table in 609-EMD-001, *Release 7 Operations Tools Manual for the EMD Project*).

6 If an error message associated with the Resource Scheduler is received, refer to Table 12.5-3, Resource Scheduler User Messages.

- The table is adapted from the corresponding table in 609-EMD-001, *Release 7 Operations Tools Manual for the EMD Project*).

7 If some other type of problem is encountered, check the appropriate log file.

- For detailed instructions refer to the **Check Log Files** procedure (Section 12.5.2).

8 If the problem cannot be identified and fixed without help within a reasonable period of time, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 12.5-2. Resource Editor User Messages (1 of 3)

Message Text	Impact	Cause and Corrective Action
A resource with this name already exists - re-enter name	Each resource name in the database must be unique.	1. Enter a different name in the **Resource Name** field. 2. **Single-click** on the **Save** button.
Activity Type is Not initialized	Without this field initialized, the “Save” operation gets rejected.	1. Shut down all Resource Planning tasks. [For detailed instructions refer to the **Shut Down Resource Definition Applications** procedure (Section 12.2.7).] 2. Notify the Database Administrator to have the PDPS database initialized (run the EcPlDbBuild script in the /usr/ecs/<MODE>/CUSTOM/utilities directory). 3. Relaunch Resource Planning applications. [For detailed instructions refer to the **Launch the Resource Editor** procedure (Section 12.2.2).] 4. Resume the operation that elicited the error message.
Block Size must be an integer number – reenter	Integer only.	1. Enter the appropriate integer (e.g., 1024) in the **Block Size** field. 2. **Single-click** on the **Save** button.
Block Size required	This is a required field.	1. Enter the appropriate integer (e.g., 1024) in the **Block Size** field. 2. **Single-click** on the **Save** button.
Error modifying computer resource	Database interface error.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 12.5-2. Resource Editor User Messages (2 of 3)

Message Text	Impact	Cause and Corrective Action
Error modifying computer resource comments	Database interface error.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Error saving computer resource	The operation failed due to an error in the database interface.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Error saving computer resource comments	Database interface error.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Number of cpus must be an integer number	Non-numeric data are not valid.	1. Enter the appropriate integer (e.g., 22) in the **Number of CPUs** field. 2. **Single-click** on the **Save** button.
Number of cpus required	This is a required field.	1. Enter the appropriate integer (e.g., 22) in the **Number of CPUs** field. 2. **Single-click** on the **Save** button.
Operating system required	This is a required field.	1. Enter the appropriate operating system data (e.g., IRIX 6.5.17) in the **Operating System** field. 2. **Single-click** on the **Save** button.
Partition Size must be a number - reenter	Integer only.	1. Enter the appropriate integer (e.g., 400000) in the **Partition Size** field. 2. **Single-click** on the **Save** button.
Partition Size required	This is a required field.	1. Enter the appropriate integer (e.g., 400000) in the **Partition Size** field. 2. **Single-click** on the **Save** button.
Resource is reserved - cannot modify	The Resource Scheduler GUI reserves the resource.	If possible, leave the resource alone. However, if the resource definition needs to be modified immediately, use the **Resource Scheduler** to change the status or delete the reservation. 1. If the resource definition needs to be modified immediately, first delete all resource reservations that specify the resource. [For detailed instructions refer to the **Delete a Resource Reservation Request** procedure (Section 12.3.8).] 2. Modify the resource definition. [For detailed instructions refer to the **Modify a Resource** procedure (Section 12.2.5).]

Table 12.5-2. Resource Editor User Messages (3 of 3)

Message Text	Impact	Cause and Corrective Action
Resource name required	Each resource in the database must have a unique name.	1. Enter an appropriate name in the **Resource Name** field. 2. **Single-click** on the **Save** button.
Resources loaded	The resources list has been loaded from the MSS baseline configuration.	For information only. No action is necessary.
Resources not loaded - file not found	The MSS baseline configuration file is not found in the previously designated directory.	No Longer Applicable.
Select a resource to modify from the list	The selected resource should be one of the defined resources.	Select (highlight) the resource to be modified in the **Resource Name** list displayed on the **Resource Editor**.
Strings should be selected	AutoSys definition requires the association of a string name.	1. Move string resources between the **Strings** and **Associated Strings** lists as necessary by selecting (highlighting) the string to be moved, then **single-clicking** on the right or left arrow button (as applicable) to move the string to the other list. 2. **Single-click** on the **Save** button.
Total ram must be an integer number	Integer only.	1. Enter the appropriate integer (e.g., 1000) representing the computer's total RAM (in megabytes) in the **Total RAM** field. 2. **Single-click** on the **Save** button.
Total ram required	This is a required field.	1. Enter the appropriate integer (e.g., 1000) representing the computer's total RAM (in megabytes) in the **Total RAM** field. 2. **Single-click** on the **Save** button.
Unable to lock Resource tables - cannot delete resource	The processing software uses the resource or its member resource.	Do not delete the resource definition at this time. The resource definition cannot be deleted while the resource is in use. 1. Wait until the resource has been released by Data Processing. 2. Try again to delete the resource definition. [For detailed instructions refer to the **Delete a Resource** procedure (Section 12.2.6).]
Unable to lock Resource tables - cannot modify resource	The processing software uses the resource or its member resource.	Do not modify the resource definition at this time. The resource definition cannot be modified while the resource is in use. 1. Wait until the resource has been released by Data Processing. 2. Try again to modify the resource definition. [For detailed instructions refer to the **Modify a Resource** procedure (Section 12.2.5).]

Table 12.5-3. Resource Scheduler User Messages (1 of 4)

Message Text	Impact	Cause and Corrective Action
A Reservation must be selected to delete	Operator cannot proceed.	1. If the desired resource reservation request is not included in the list displayed on the **Resource Scheduler**, **single-click** and **hold** on the **Activity Type** option button and select the appropriate category of activity (or select **All**) from the option menu that is displayed. 2. **Single-click** on the resource reservation request to be deleted. 3. Execute **File** → **Delete** from the **Resource Scheduler** pull-down menu.
A Reservation must be selected to modify	Operator cannot proceed.	1. If the desired resource reservation request is not included in the list displayed on the **Resource Scheduler**, **single-click** and **hold** on the **Activity Type** option button and select the appropriate category of activity (or select **All**) from the option menu that is displayed. 2. From the **Resource Scheduler**, **single-click** on the resource reservation request to be modified. 3. **Single-click** on the **Modify…** button to access the **Resource Reservation Request Edit/Definition** GUI.
Can't insert new ResvName: <name> into database	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
can't send requestActChg to resource model for resvName: <name>	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Delete ResvName: <name> from the list	The database cannot be updated.	Delete the resource reservation request. [For detailed instructions refer to the **Delete a Resource Reservation Request** procedure (Section 12.3.8).]
Error in creating a new object for row: <row>.	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 12.5-3. Resource Scheduler User Messages (2 of 4)

Message Text	Impact	Cause and Corrective Action
Fail to modify resvName: <name>	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
I can't find <Plan Name>.	There is a problem with the resource pool.	Enter a valid Plan Name.
Must select reservation	Operator cannot proceed.	1. If the desired resource reservation request is not included in the list displayed on the **Resource Scheduler**, **single-click** and **hold** on the **Activity Type** option button and select the appropriate category of activity (or select **All**) from the option menu that is displayed. 2. From the **Resource Scheduler**, **single-click** on the desired resource reservation request.
New Resvation can't leave resources list of ResvName: <name> empty	This required field must be filled.	Ensure that there is at least one entry in the **Selected Resources** list on the **Resources Selection** GUI. [For detailed instructions refer to the **Selecting Resources...** procedure (Section 12.3.2.1).]
Open one Reservation at a time, Please	Reservation cannot be opened.	1. Find and **single-click** on the open **Resource Reservation Edit/Definition** GUI that is already open. 2. **Single-click** on either the **Save** or **Cancel** button (as appropriate). 3. Open the desired resource reservation request. [For detailed instructions refer to either the **Create a Resource Reservation Request** procedure (Section 12.3.2) or the **Edit a Resource Reservation Request** procedure (Section 12.3.3).]
PlRpSiScheduler::modifyReservation – can't save new info for resvName: <name>.	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
ResvName: < > already has status <status>.	Action cannot be completed.	Choose a different action.
ResvName: <name> can't replace new Interval List	The database cannot be updated.	1. Type a new (unique) name for the resource request in the **Request Name** field (**Resource Reservation Request Edit/Definition** GUI). 2. Click on the **Save** button. [For detailed instructions refer to r the **Create a Resource Reservation Request** procedure (Section 12.3.2).]

Table 12.5-3. Resource Scheduler User Messages (3 of 4)

Message Text	Impact	Cause and Corrective Action
ResvName: <name> Selected Intervals list can't be empty	This required field must be filled.	Ensure that there is at least one entry in the **Selected Intervals** list on the **Intervals Selection** GUI. [For detailed instructions refer to the **Deselecting Intervals...** procedure (Section 12.3.3.1).]
ResvName: <name> Selected Resources list can't be empty	This required field must be filled.	Ensure that there is at least one entry in the **Selected Resources** list on the **Resources Selection** GUI. [For detailed instructions refer to the **Selecting Resources...** procedure (Section 12.3.2.1).]
ResvName: <name> accepts new resources list	Informational message.	For information only. No action is necessary.
resvName: <name> can't replace new Resource List	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
ResvName: <name> can't uncommitted < > RActAlls	The database cannot be updated.	1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 12.5.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
resvName: <name> fails to approve - status is changed to <status>.	The plan cannot be approved due to conflicts with other reservations.	Resolve the conflict(s). [For detailed instructions refer to the **Approve a Resource Reservation Request** procedure (Section 12.3.5).]
resvName: <name> myTime: <time> resourceName: <name> conflicted Time: <time> conflictResvName: <name>	Informational message.	For information only. No action is necessary.
ResvName: <name> resource list is less now	Informational message.	For information only. No action is necessary.
ResvName: <name> status is changed from approved to committed	Informational message.	For information only. No action is necessary.
ResvName: <name> status is changed to <status>	Informational message.	For information only. No action is necessary.

Table 12.5-3. Resource Scheduler User Messages (4 of 4)

Message Text	Impact	Cause and Corrective Action
Success to approve reservation Name: <name>	Informational message.	For information only. No action is necessary.
Success to update resvName: <name> name	Informational message.	For information only. No action is necessary.
This Name: <name> with status: <status> has been used, Please pick another Name.	The resource reservation request name must be unique.	1. Enter a different name in the **Request Name** field. 2. **Single-click** on the **Save** button.

12.5.2 Check Log Files

Log files can provide indications of the following types of problems:

- Communication problems.
- Database problems.
- Lack of disk space.

Table 12.5-4 presents (in a condensed format) the steps required to check log files. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Access a terminal window logged in to the appropriate host.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 12.2.1).

2 At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- **<MODE>** is current mode of operation.
  - TS1 - Science Software Integration and Test (SSI&T)
  - TS2 - New Version Checkout
  - OPS - Normal Operations
- “logs” is the directory containing resource planning log files (e.g., EcPlRpRe.ALOG, EcPlRpReDebug.log, EcPlRpSi.ALOG, or EcPlRpSiDebug.log).

3 At the command line prompt enter:

pg <file name>

- **<file name>** refers to the resource planning log file to be reviewed (e.g., EcPlRpRe.ALOG, EcPlRpReDebug.log, EcPlRpSi.ALOG, or EcPlRpSiDebug.log.
- The first page of the log file is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **vi**, **view**) can be used to review the log file.

4 Review the log file to identify problems that have occurred.

- To exit from **pg** at the **:** prompt enter:

  q

  – The command line prompt is displayed.

5 Respond to problems as follows:

- Resource Planning-related problems.
  – Perform the appropriate procedure(s) from Table 12.5-1, Troubleshooting Resource Planning Problems.
- Communication problems.
  – Notify the Operations Controller/System Administrator of suspected communication problems.
- Database problems.
  – Verify that relevant database servers are running.
  – Check for lack of (or corruption of) data in the database using either a database browser or interactive structured query language (isql) commands.
  – Notify the Database Administrator of suspected database problems.
- Lack of disk space.
  – Remove unnecessary files.
  – Notify the Operations Controller/System Administrator of recurring disk space problems.

Table 12.5-4. Check Log Files - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window	**single-click** or use procedure in Section 12.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/logs**	**enter text, press Enter**
3	**pg <file name>**	**enter text, press Enter**
4	identify problems indicated in the log file	**read text**
5	Respond to problems as necessary	

12.5.3 Check Database Connections

If applications (including the GUIs) are unable to connect to the database, data cannot be retrieved or (in the case of the GUIs) displayed. Consequently, if a GUI does not display data or if the display does not refresh, checking the database connections is a logical step in trying to isolate the problem.

Table 12.5-5 presents (in a condensed format) the steps required to check database connections. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Submit a request to the Database Administrator to identify the values for parameters associated with the appropriate application.

- The following parameters should be requested:
  - **DBName.**
  - **DBServer.**
  - **DBMaxConnections.**
- The preceding parameters are associated with the following applications:
  - EcPlRpRe.
  - EcPlRpRm.
  - EcPlRpSi.
  - EcPlRpTl.

2 Access a terminal window logged in to the Queuing Server host.

- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03.**
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 12.2.1).

3 At the command line prompt enter:

isql –U <user ID> -S <database server>

- **<user ID>** is the database user's identification; e.g., **pdps_role**.
- **<database server>** is the database server; e.g., **g0sps06_srvr**.

4 At the **Password:** prompt enter:

<database password>

- **<database password>** is the password for logging in to the database using the specified **<user ID>**.
- A **1>** prompt is displayed, indicating that a connection has been made with the database.

5 At the **1>** prompt enter:

sp_who

6 At the **2>** prompt enter:

go

- A listing of connections to the database is displayed.
- The listing includes data in the following columns:
  - **spid.**
  - **status.**
  - **loginame.**
  - **hostname.**
  - **blk.**
  - **dbname.**
  - **cmd.**

7 At the **1>** prompt enter:

sp_configure "user connections"

8 At the **2>** prompt enter:

go

- A listing of connections to the database is displayed.
- The listing includes the following types of data:
  - **Parameter Name** (i.e., number of user connections).
  - **Default.**
  - **Memory Used.**
  - **Config Value.**
  - **Run Value.**

9 To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.

10 Compare the number of actual connections (results of **sp_who**) with the number of connections for which the database has been configured (results of **sp_configure "user connections"**).

11 If the number of actual connections is very close to the number of connections for which the database has been configured, notify the Database Administrator of the fact.

12 If the number of actual connections is **not** very close to the number of connections for which the database has been configured, compare the number of actual connections with the value for DBMaxConnections that the Database Administrator specified (Step 1).

13 If the number of actual connections is very close to the value for DBMaxConnections, notify the Database Administrator of the fact.

- It may be advisable to increase the value assigned to the DBMaxConnections parameter in the Configuration Registry.

Table 12.5-5. Check Database Connections - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Identify the values for database parameters associated with the appropriate Resource Planning application	**contact Database Administrator**
2	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 12.2.1
3	**isql –U <user ID> -S <database server>**	**enter text, press Enter**
4	**<database password>**	**enter text, press Enter**
5	**sp_who**	**enter text, press Enter**
6	**go**	**enter text, press Enter**
7	**sp_configure "user connections"**	**enter text, press Enter**
8	**go**	**enter text, press Enter**
9	**quit**	**enter text, press Enter**
10	Compare the number of actual connections with the number of connections for which the database has been configured	**read text**
11	Notify the Database Administrator of the results	**contact Database Administrator**

This page intentionally left blank.

13. Production Planning

13.1 Production Planning Process

The Planning Subsystem supports site operations in developing a production plan based on locally defined strategy. Production Planning involves creation of Production Requests using the Production Request Editor and the scheduling of the Production Requests using the Planning Workbench.

The Production Planner defines ECS science data processing requirements in terms of Production Requests (PRs).

- A PR is an order for data to be produced by the Data Processing Subsystem.
- Each PR identifies specific science software [in the form of a Product Generation Executive (PGE)] that is to be run and the data (in terms of a time period, set of orbits, etc.) that is to be processed to produce the desired output product(s).

In response to a PR the Planning Subsystem generates either one or a series of Data Processing Requests (DPRs).

- Each DPR corresponds to one execution of a single PGE.
- Each DPR contains the information that is needed by the Data Processing Subsystem to execute the PGE and produce the desired output product(s).

The Production Planner defines Production Strategies that specify values in several categories related to DPRs.

- The values specified in a Production Strategy can be used in determining the priority of each DPR.

The Production Planner creates a Production Plan that specifies which PR(s) [and consequently which DPR(s)] should be sent to Data Processing at one time.

- When creating a Production Plan, the Production Planner can associate a particular Production Strategy with the PRs in the plan.
- When the Production Planner "activates" a Production Plan, the associated DPRs become accessible to Data Processing.

The Production Planner uses the following principal tools in the Planning Subsystem:

- Production Request Editor GUI - for creating Production Requests and Data Processing Requests.
- Production Strategies GUI - for creating Production Strategies.
- Planning Workbench GUI – for creating and activating Production Plans.
- Planning Timeline – for reviewing Production Plans.

Subsequent sections related to Production Planning address the following topics:

- Section 13.2 An overview of the process for creating or deleting Production Requests and Data Processing Requests and step-by-step procedures for using the Production Request Editor.
- Section 13.3 An overview of the process for creating, modifying, or deleting Production Strategies and step-by-step procedures for using the Production Strategies GUI.
- Section 13.4 An overview of the process for creating, activating, or deleting Production Plans and step-by-step procedures for using the Planning Workbench GUI and Planning Timeline.
- Section 13.5 An overview and step-by-step procedures for resetting/cleaning the PDPS database.
- Section 13.6 Referral to the procedures for tuning system parameters included in Chapter 14, Production Processing.
- Section 13.7 An overview of the process and step-by-step procedures for troubleshooting Production Planning problems.

Data Preprocessing (DPREP)

DPREP (data preprocessing) consists of sets of PGEs that use a statistical approach to convert Level 0 (L0) ephemeris and attitude ancillary data for a particular satellite (e.g., Terra or Aqua) into SDP Toolkit native binary format without altering or modifying the scientific content of the granules.

DPREP PGEs are supplied by ECS, unlike most PGEs, which are provided by the Science Computing Facilities that ECS supports. Release 7 DPREP supports Terra and Aqua operations and will eventually support Aura operations.

Terra DPREP

Terra DPREP consists of the following three PGEs:

- AM1Eph or Step 1 (EcDpPrAm1EdosEphAttDPREP_PGE), a ksh script that serves as a driver for the following three executables:
  - EcDpPrAm1EdosAncillary
  - EcDpPrAm1EdosEphemerisRepair
  - EcDpPrAm1ToolkitToHdf
- FddAtt or Step 2 (EcDpPrAm1FddAttitudeDPREP_PGE).
- RepEph or Step 3 (EcDpPrAm1FddEphemerisDPREP_PGE).

Operationally, Steps 1 and 2 are scheduled daily and run independently of one another. Step 3 is scheduled and run on an as-needed basis.

There are several sources of information on the Terra DPREP PGEs and how to run them:

- 500-EMD-001, Terra Spacecraft Ephemeris and Attitude Data Preprocessing.

- 611-EMD-001, Mission Operation Procedures for the EMD Project, Chapter 26.
- Two files installed on the science processor hosts (e.g., e0spg11, g0spg11, or l0spg11) in the /usr/ecs/*MODE*/CUSTOM/data/DPS directory.
  - "AM1_DPREP_README"
  - "HowToRunAm1DPREP"

The Terra DPREP PGEs process Level 0 Terra (AM-1) spacecraft data (e.g., ESDT AM1ANC) provided by EDOS. The output files/granules of the DPREP PGEs are subsequently used in the processing of data from various instruments on the satellite. They provide the following types of ancillary (non-science) data:

- Ephemeris.
  - Spacecraft location: ephemeris (or orbit) data include: latitude, longitude, and height.
- Attitude.
  - Orientation of the satellite, including yaw, pitch, and roll angles; and angular rates about three axes.

AM1Eph (Step 1 DPREP: EDOS Level 0 Ancillary Data)

EcDpPrAm1EdosEphAttDPREP_PGE consists of a script that coordinates the following three DPREP executables:

- EcDpPrAm1EdosAncillary.
- EcDpPrAm1EdosEphemerisRepair.
- EcDpPrAm1ToolkitToHdf.

EcDpPrAm1EdosAncillary reads in an AM-1 L0 (EDOS) ancillary granule (ESDT AM1ANC). It also reads another AM1ANC granule. The second granule is required to ensure that incomplete orbits in the first granule get complete orbit metadata records. (The only data that is extracted from the second granule is the descending node time and longitude.) EcDpPrAm1EdosAncillary also reads in ephemeris and attitude data (ESDT AM1EPHN0 and ESDT AM1ATTN0). These would be the last ephemeris/attitude granules generated from a previous run of the PGE.

If EcDpPrAm1EdosAncillary signals that a short gap was detected, EcDpPrAm1EdosEphemerisRepair reads the scratch file created by EcDpPrAm1EdosAncillary, fills the gap, and writes a gap-filled native-format (Toolkit-format) ephemeris file.

NOTE: EDOS-supplied ephemeris data are the primary source of ephemeris for Terra. However, EDOS-supplied attitude data are not the primary source of attitude for Terra. Attitude data supplied by the Flight Dynamics Division (FDD) are considered the primary source of Terra attitude.

EcDpPrAm1EdosAncillary performs a full complement of data quality analyses on the EDOS ephemeris data. In contrast EDOS-supplied attitude data are subject to minimal quality checks and never undergo data repair because EDOS-supplied attitude data are not considered good enough for science data processing. EDOS attitude data are "use at own risk" data; the data recommended for science data processing are the FDD attitude data routinely preprocessed by Step 2 DPREP (EcDpPrAm1FddAttitudeDPREP_PGE).

EcDpPrAm1ToolkitToHdf takes the native format ephemeris file and produces a corresponding HDF file and a metadata file.

EcDpPrAm1EdosEphAttDPREP_PGE produces Toolkit- and HDF-format attitude (ESDTs AM1ATTN0 and AM1ATTH0) and ephemeris (ESDTs AM1EPHN0 and AM1EPHH0) granules.

NOTE: The Level 0 (input) granules represent two-hour periods of time but processing cannot be performed on the most recently available two-hour granule. Step 1 processing needs to look forward in the time stream in order to complete orbit metadata processing.

FddAtt (Step 2 DPREP: Attitude Data)

EcDpPrAm1FddAttitudeDPREP_PGE reads in both the current FDD attitude granule (AM1ATTF) and the next FDD attitude granule. It also reads in the attitude granule (AM1ATTNF) it produced with its last run. The output of the process is a native-format attitude file (AM1ATTNF) and an HDF-format attitude file (AM1ATTHF). A metadata file is produced for each output data file.

RepEph (Step 3 DPREP: Repaired Ephemeris Data)

If Step 1 finds too many missing data points in the ephemeris data (e.g., AM1EPHH0 and AM1EPHN0 granules have gaps of greater than 60 seconds), it signals that there are problems with the data. The Production Planner receives an email message indicating the problems in the preprocessed Terra ephemeris granules and notifies the FDD (e.g., by telephone) that a repaired ephemeris file (AM1EPHF) is needed for the time span of the granule that has the gap.

When the repaired ephemeris file has been ingested, EcDpPrAm1FddEphemerisDPREP_PGE reads in the repaired ephemeris granule and the EOS_AM1 ephemeris data in Toolkit native format. It produces replacement ephemeris granules (AM1EPHH0 and AM1EPHN0).

Terra DPREP Profiles

As previously mentioned DPREP processing has data requirements beyond the current two-hour segment. Data from the preceding and following segments are used for performing consistency checks on the ephemeris and attitude data streams when the data streams bridge segment boundaries. However, there is no guarantee that data from the preceding and following segments will always be available. Consequently, four data processing profiles have been developed for each of the three DPREP steps to accommodate the various permutations of data availability:

- Profile 1 is used when data are available from the preceding, current, and following segments.

- Profile 2 is used when data are available from the current and following segments only.
- Profile 3 is used when data are available from the preceding and current segments only.
- Profile 4 is used when data are available from the current segment only.

Profile 1 is used for nominal DPREP operation. It is the profile of each DPREP step that is run on a routine basis.

Profile 2 (no preceding data, but following data is available) initializes DPREP's processing of a given step's ephemeris and/or attitude data stream. When Profile 2 has been run on a data segment, Profile 1 (preceding and following data available) assumes processing responsibility on all data segments thereafter until data dropout or mission end is encountered.

Profile 3 (preceding data available, but no following data) processes the data segment that immediately precedes data dropout and, therefore, terminates processing on a given step's ephemeris and/or attitude data stream.

Profile 4 is used for processing isolated data segments and is not likely to be scheduled operationally.

In the big picture of the mission, DPREP processing on the very first data segment would require running Profile 2. The next data segments would be processed using Profile 1 processes. The very last data segment of the mission could be processed using Profile 3. However, PDPS never halts DPREP processing if a following granule does not become available. So if a following granule does not become available after a four-hour wait period, DPREP (Profile 1) reverts to Profile 3 processing, which does not depend on the raw input granule from the segment that follows. Given the processing flexibility of Profile 1, neither the scheduling of Profile 3 nor the scheduling of Profile 4 is envisioned operationally.

Aqua DPREP

Aqua DPREP consists of the following processes:

- PM1DefEph (EcDpPrPm1FddEphemerisDPREP_PGE) Aqua Ephemeris Processing.
- PM1DefAtt (EcDpPrPm1AttitudeDPREP_PGE) Aqua Attitude Processing.

There are several sources of information on the Aqua DPREP PGEs and how to run them:

- 500-EMD-002, Aqua Spacecraft Ephemeris and Attitude Data Preprocessing.
- 611-EMD-001, Mission Operation Procedures for the EMD Project, Chapter 26.
- "HowToRunPm1DPREP" file installed on the science processor hosts (e.g., e0spg11, g0spg11, or l0spg11) in the /usr/ecs/MODE/CUSTOM/data/DPS directory.

Aqua DPREP includes the processing of both ephemeris and attitude data:

- Ephemeris data is received from the FDD in ephemeris data files.
  - Ephemeris data arrives at the DAAC daily about eight to ten hours after the end of the UTC day.

- The Ground-Based Attitude Determination (GBAD) data that is used in the processing of attitude data is received from the ECS Mission Operations Segment (EMOS) in "carry-out" files.
  - GBAD carry-out files (PMCOGBAD) are used in conjunction with ephemeris data to prepare attitude data.

Operationally, Ephemeris and Attitude PGEs are scheduled daily and run independently of one another. However, Ephemeris is always run first on any given data segment because attitude processing depends on ephemeris data to complete its processing.

PM1DefEph (Aqua Ephemeris Processing)

Aqua ephemeris processing includes reformatting FDD ephemeris granules into Toolkit native format and HDF format. In addition, ephemeris metadata records are generated for the product granules.

The Aqua Ephemeris DPREP PGE processes Aqua ancillary granules (i.e., PM1EPHD, ephemeris data) supplied by FDD. The output files/granules of the Aqua Ephemeris DPREP PGE provide satellite ephemeris data and are subsequently used in the processing of Aqua satellite attitude data.

The Aqua Ephemeris DPREP PGE reads in both the current PM1EPHD ephemeris granule for Aqua and a previous PM1EPHND preprocessed Aqua platform ephemeris granule in native format. The outputs of the process are preprocessed Aqua platform ephemeris granules in native format (PM1EPHND) and HDF format (PM1EPHHD). A metadata file is produced for each output data file.

PM1DefAtt (Aqua Attitude Processing)

Aqua Attitude DPREP inputs are from ESDTs PMCOGBAD and PM1EPHND (preprocessed Aqua platform ephemeris data in native format). The outputs of Aqua Attitude DPREP are PM1ATTHR (Aqua attitude data in HDF-EOS format) and PM1ATTNR (Aqua attitude data in native format).

The attitude DPREP stream executes (nominally) twelve times per day at two-hour intervals. The PMCOGBAD data is in the form of two-hour granules (12 granules per day). Each PM1EPHND granule represents 12 hours of data (two granules per day).

Interpretation of the attitude data depends on the value of the Guidance, Navigation, and Control (GN&C) Status Word 2 contained in the carry-out file GBAD data. In general if the value of GN&C Status Word 2 represents "fine point" mode, the attitude is acceptable for Aqua science processing. Any other value indicates inadequate attitude accuracy for Aqua science data processing.

EcDpPrPm1AttitudeDPREP_PGE reads in the previous, current, and next FDD attitude granules (PMCOGBAD). It reads in the ephemeris (PM1EPHND) granule for the same time period (and adjacent granule if near a granule boundary). It also reads in the attitude granule (PM1ATTNR) produced during the previous run. The output of the process is a native-format attitude file (PM1ATTNR) and an HDF-format attitude file (PM1ATTHR). A metadata file is produced for each output data file.

Aqua Data Repair

Because FDD and EMOS have preprocessed the ephemeris and attitude data streams, data repair has been deemed unnecessary and is not performed.

Aqua DPREP Profiles

As mentioned with regard to Terra DPREP processing Aqua DPREP data requirements extend beyond the current two-hour segment. Data from the preceding and following segments are used in performing consistency checks on the ephemeris and attitude data streams when the data streams bridge segment boundaries. However, there is no guarantee that data from the preceding and following segments will always be available. Consequently, four data processing profiles have been developed for each of the Aqua DPREP PGEs to accommodate the various permutations of data availability. The Aqua DPREP profiles have the same functions as the corresponding Terra DPREP profiles.

Aura DPREP

Operationally, Aura DPREP is very similar to Aqua DPREP. Aura DPREP consists of the following processes:

- AuraEph (EcDpPrAuraEphemerisDPREP_PGE) Aura Ephemeris Processing.
- AuraAtt (EcDpPrAuraAttitudeDPREP_PGE) Aura Attitude Processing.

There are several sources of information on the Aura DPREP PGEs and how to run them:

- 500-EMD-003, Aura Spacecraft Ephemeris and Attitude Data Preprocessing.
- 611-EMD-001, Mission Operation Procedures for the EMD Project, Chapter 26.
- “HowToRunAuraDPREP” file installed on the science processor hosts (e.g., e0spg11, g0spg11, or l0spg11) in the /usr/ecs/MODE/CUSTOM/data/DPS directory.

Aura DPREP includes the processing of both ephemeris and attitude data:

- Ephemeris data is received from the FDD in ephemeris data files.
  - Ephemeris data arrives at the DAAC daily about eight to ten hours after the end of the UTC day.
- The Ground-Based Attitude Determination (GBAD) data that is used in the processing of attitude data is received from EMOS in "carry-out" files.
  - GBAD carry-out files (AUCOGBAD) are used in conjunction with ephemeris data to prepare attitude data.

Operationally, Ephemeris and Attitude PGEs are scheduled daily and run independently of one another. However, Ephemeris is always run first on any given data segment because attitude processing depends on ephemeris data to complete its processing.

AuraEph (Aura Ephemeris Processing)

Aura ephemeris processing includes reformatting FDD ephemeris granules into Toolkit native format and HDF format. In addition, ephemeris metadata records are generated for the product granules.

The Aura Ephemeris DPREP PGE processes Aura ancillary granules (i.e., AUREPHMF, ephemeris data) supplied by FDD. The output files/granules of the Aura Ephemeris DPREP PGE provide satellite ephemeris data and are subsequently used in the processing of Aura satellite attitude data.

The Aura Ephemeris DPREP PGE reads in both the current AUREPHMF ephemeris granule for Aura and a previous AUREPHMN preprocessed Aura platform ephemeris granule in native format. The outputs of the process are preprocessed Aura platform ephemeris granules in native format (AUREPHMN) and HDF format (AUREPHMH). A metadata file is produced for each output data file.

AuraAtt (Aura Attitude Processing)

Aura Attitude DPREP inputs are from ESDTs AUCOGBAD and AUREPHMN (preprocessed Aura platform ephemeris data in native format). The outputs of Aura Attitude DPREP are AURATTH (Aura attitude data in HDF-EOS format) and AURATTN (Aura attitude data in native format).

The attitude DPREP stream executes (nominally) twelve times per day at two-hour intervals. The AUCOGBAD data is in the form of two-hour granules (12 granules per day). Each AUREPHMN granule represents 12 hours of data (two granules per day).

Interpretation of the attitude data depends on the value of the Guidance, Navigation, and Control (GN&C) Status Word 2 contained in the carry-out file GBAD data. In general if the value of GN&C Status Word 2 represents "fine point," “attitude hold,” or “earth point” mode, the attitude is acceptable for Aura science processing. Any other value indicates inadequate attitude accuracy for Aura science data processing.

EcDpPrAuraAttitudeDPREP_PGE reads in the previous, current, and next FDD attitude granules (AUCOGBAD). It reads in the ephemeris (AUREPHMN) granule for the same time period (and adjacent granule if near a granule boundary). It also reads in the attitude granule (AURATTN) produced during the previous run. The output of the process is a native-format attitude file (AURATTN) and an HDF-format attitude file (AURATTH). A metadata file is produced for each output data file.

Aura Data Repair

Because FDD and EMOS have preprocessed the ephemeris and attitude data streams, data repair has been deemed unnecessary and is not performed.

Aura DPREP Profiles

As mentioned with regard to Terra and Aqua DPREP processing Aura DPREP data requirements extend beyond the current two-hour segment. Data from the preceding and following segments are used in performing consistency checks on the ephemeris and attitude data streams when the data streams bridge segment boundaries. However, there is no guarantee that data from the preceding and following segments will always be available. Consequently, four data processing profiles have been developed for each of the Aura DPREP PGEs to accommodate the various permutations of data availability. The Aura DPREP profiles have the same functions as the corresponding Terra DPREP profiles.

13.2 Creating/Deleting Production Requests and Data Processing Requests

From the Production Request Editor, the Production Planner can create new production requests, modify or delete production requests, and review or delete data processing requests. In addition, the Production Request (PR) Generator (EcPlPRGenerator) provides the Production Planner with a means of creating new production requests using a command-line interface (not a GUI).

Each procedure outlined has an **Activity Checklist** table that provides an overview of the task to be completed. The outline of the **Activity Checklist** is as follows:

Column one - ***Order*** shows the order in which tasks should be accomplished.

Column two - ***Role*** lists the Role/Manager/Operator responsible for performing the task.

Column three - ***Task*** provides a brief explanation of the task.

Column four - ***Section*** provides the Procedure (P) section number or Instruction (I) section number where details for performing the task can be found.

Column five - ***Complete?*** is used as a checklist to keep track of which task steps have been completed.

Table 13.2-1 provides an Activity Checklist for activities related to the creation of Production Requests and Data Processing Requests.

Table 13.2-1. Production Requests and Data Processing Requests - Activity Checklist (1 of 2)

Order	Role	Task	Section	Complete?
1	Production Planner	Log in to ECS Hosts	(P) 13.2.1	
2	Production Planner	Launch the Production Request Editor	(P) 13.2.2	
3	Production Planner	Create a New Production Request Using the Production Request Editor GUI	(P) 13.2.3	

Table 13.2-1. Production Requests and Data Processing Requests - Activity Checklist (2 of 2)

Order	Role	Task	Section	Complete?
4	Production Planner	Create New Production Requests Using the Production Request Generator (Command-Line Interface)	(P) 13.2.4	
5	Production Planner	Edit/Modify a Production Request	(P) 13.2.5	
6	Production Planner	Delete a Production Request	(P) 13.2.6	
7	Production Planner	Review Data Processing Requests	(P) 13.2.7	
8	Production Planner	Delete a Data Processing Request	(P) 13.2.8	
9	Production Planner	Re-Generate Granules Affected by Loss of Files from the Archive	(P) 13.2.9	

NOTE: The procedures that follow are written under the assumption that PGEs have been previously created and are available for use with the same rule criteria that you are attempting to use.

13.2.1 Log in to ECS Hosts

Logging in to ECS hosts is accomplished from a UNIX command line prompt. Table 13.2-2 presents (in a condensed format) the steps required to log in to ECS hosts. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1 At the UNIX command line prompt enter:

setenv DISPLAY <client name>:0.0

- Use either the X terminal/workstation IP address or the machine-name for the client name.
- When using secure shell, the DISPLAY variable is set just once, before logging in to remote hosts. If it were to be reset after logging in to a remote host, the security features would be compromised.

2 In the terminal window (at the command line prompt) start the log-in to the appropriate host by entering:

/tools/bin/ssh <host name>

- The **-l** option can be used with the ssh command to allow logging in to the remote host (or the local host for that matter) with a different user ID. For example, to log in to x0pls02 as user cmops enter:

  /tools/bin/ssh -l cmops x0pls02

- Depending on the set-up it may or may not be necessary to include the path (i.e., /tools/bin/) with the ssh command. Using ssh alone is often adequate. For example:

  ssh x0pls02

  - or -

  ssh -l cmops x0pls02
- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- Examples of Science Processor host names include **e0spg11**, **g0spg11,** and **l0spg11**.
- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03**.
- Examples of Access/Process Coordinators (APC) Server host names include **e0acg11**, **g0acg01**, and **l0acg02**.
- Examples of Ingest Server host names include **e0icg11, g0icg01**, and **l0acg02**.
- Examples of Sun external server host names include **e0ins01, g0ins01**, and **l0ins01**.
- Examples of Sun internal server host names include **e0acs06**, **g0acs06**, and **l0acs06**.
- If you receive the message, "Host key not found from the list of known hosts. Are you sure you want to continue connecting (yes/no)?" enter **yes** ("y" alone will not work).
- If you have previously set up a secure shell passphrase and executed **sshremote**, a prompt to **Enter *passphrase* for RSA key '<*user@localhost*>'** appears; continue with Step 3.
- If you have not previously set up a secure shell passphrase, go to Step 4.

3 If a prompt to **Enter passphrase for RSA key '<*user@localhost*>'** appears, enter:

<passphrase>

- If a command line prompt is displayed, log-in is complete.
- If the passphrase is unknown, press **Return/Enter**, which should cause a **<user@remotehost>'s password:** prompt to appear (after the second or third try if not after the first one), then go to Step 4.
- If the passphrase is entered improperly, a **<user@remotehost>'s password:** prompt should appear (after the second or third try if not after the first one); go to Step 4.

4 If a prompt for **<*user@remotehost*>'s password:** appears, enter:

<password>

- A command line prompt is displayed.
- Log-in is complete.

Table 13.2-2. Log in to ECS Hosts - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	**setenv DISPLAY <client name>:0.0**	**enter text, press Enter**
2	**/tools/bin/ssh <host name>** (as applicable)	**enter text, press Enter**
3	**<passphrase>** (if applicable)	**enter text, press Enter**
4	**<password>** (if applicable)	**enter text, press Enter**

13.2.2 Launch the Production Request Editor

The Production Request Editor is invoked from a UNIX command line prompt. Table 13.2-3 presents (in a condensed format) the steps required to launch the Production Request Editor. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Access a terminal window logged in to the Planning/Management Workstation host.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 In the terminal window, at the command line, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- <**MODE**> is current mode of operation.
  - TS1 - Science Software Integration and Test (SSI&T)
  - TS2 - New Version Checkout
  - OPS - Normal Operations
- "utilities" is the directory containing the Planning Subsystem start-up scripts.

3 Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

4 Start the Production Request Editor by entering:

EcPlPRE_IFStart <MODE>

- The **Production Request Editor** is launched.

Table 13.2-3. Launch the Production Request Editor - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	Set environment variables if necessary	**enter text, press Enter**
4	**EcPlPRE_IFStart <MODE>**	**enter text, press Enter**

13.2.3 Create a New Production Request Using the Production Request Editor GUI

The process of creating a new Production Request using the Production Request Editor GUI begins with the Production Planner starting the Production Request Editor GUI. The Production Planner specifies the PGE, duration, and comments for the new Production Request.

TYPES OF DATA PROCESSING: ECS accommodates four general types of data processing.

Routine Processing	Pre-defined software production processing that is periodic and keyed to data arrival. For example, every day a Production Planner includes in the daily schedule a DPR for generating a particular Level 1A product from the most recent Level 0 data from the applicable satellite instrument.
Reprocessing	Typically involves using a new, improved PGE to process data that had previously been processed with an older version of the PGE.
Regeneration	A type of reprocessing that is performed for the purpose of replacing a missing or damaged product. It is necessary when an existing product has been corrupted or deleted. If the damaged or missing product is needed for shipping or as input for additional processing, it must be recreated. To the extent possible the regenerated file is created using the same input, the same processing parameters, and the same algorithm as the original file.
On-Demand Processing	Involves ad-hoc processing initiated by an end-user (as opposed to the Production Planner). For example, a researcher using data from the Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) instrument on the Terra satellite may need a particular Level 2 product that has not yet been generated. The ASTER researcher would submit an on-demand request to

have the product generated from a Level 1B product stored in the archive.

NOTE: When reprocessing, it is important to generate DPRs for one chain completely before generating any DPRs for the next chain. For example, if PGE1 produces Product1 that is input to PGE2 that produces Product2 that is input to PGE3 that produces Product3, PRs for PGE1, PGE2, and PGE3 should be created for each of those DPRs for Chain1. Then PRs for Chain2 can be created, etc.

A reprocessing DPR selects the latest version granule in the database as input. Consequently, if Product1 from Chain2 were in the database before the DPR for Chain1 PGE2 was created, the Chain1 DPR for PGE2 would select the Product1 from Chain2 as its input and Product1 from Chain1 would not be used.

PRODUCTION RULES: Production Rules provide templates for Instrument Teams to describe the relationship(s) between PGEs and their input and output data. The assumption of this documentation is that the user has knowledge of the specific production rules under which the PGE was created. Listed below is a sampling of the available production rules.

Basic Temporal	Temporal (time) range of inputs matches the temporal range of outputs.
Advanced Temporal	Temporal range of inputs is offset from the expected temporal range of inputs and outputs.
Alternate Inputs	PGE is run with different inputs based on the availability of various alternate input data sets.
Optional Inputs	PGE is run with specified optional inputs if available; otherwise, PGE is run without them.
Min/Max Granules	Minimum number of input granules needed for full data coverage and maximum number of input granules to search for may be specified. Minimum and maximum number of outputs expected from the PGE may be specified.
Optional DPRs	The only DPRs executed are those for which the non-routine key input data actually become available (i.e., are either produced in data processing or can be acquired from the archive).
Metadata Checks	DPR is run only if input data’s metadata value(s) meet(s) certain criteria.
Metadata Query	Input granule selection is based on metadata value(s).
Spatial Query/Spatial Pad	Input granule selection is based on the spatial coverage of another input (i.e., the key input). Spatial Pad involves adding area to all sides of the key input's spatial shape. All granules that intersect the expanded area are retrieved.
Closest Granule	DPR is generated if a required input granule within a particular time range (rather than an exact time) is available; otherwise, no DPR is generated.
Orbital Processing	Selection of input times is based on orbit information.
Multiple DPRs for Insertion Time	Allows the creation of DPRs for multiple granules with the same insertion time (affects ASTER L1B routine processing only).
Tiling	Input data is chosen on the basis of Instrument Team-defined tiles (geographic areas).

Table 13.2-4 presents (in a condensed format) the steps required to create a new production request using the Production Request Editor GUI. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Launch the **Production Request Editor** (refer to Section 13.2.2).

- The **Production Request Editor** GUI is displayed.

ASTER *routine processing makes use of the* ***Optional DPRs Production Rule*** *to schedule and execute ASTER PGEs for new data that have been archived. The Production Planner specifies the* ***insertion time*** *range (i.e., the time period when the desired data were archived) as opposed to the* ***collection time*** *(when the satellite instrument gathered the data).*

2 **Single-click** on the **PR Edit** tab.

- The **PR Edit** GUI is displayed.

NOTE: If the GUI is unresponsive, always check to see if a prompt window is hidden behind the main GUI waiting for a response. Respond to the window, then continue with the Production Request Editor GUI.

NOTE: The **Help** buttons at the bottom of various the **PR Edit GUI** pages are non-functional (do not work).

3 **Single-click** and **hold** the **PR Type** option button to display a menu of types of production requests, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.

- Options are: **Routine, On-Demand** [not currently available for selection], and **Reprocessing.**

4 **Single-click** and **hold** the **User Type** option button to display a menu of types of users, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.

- Options are: **Operator, DAAC Manager, Scientist,** and **Researcher.**

NOTE: The **PR Name** and **Origination Date** fields will be filled automatically when the Production Request is saved at the end of the procedure. (You do not need to fill in these fields.)

5 In the **Originator** field either enter the actual name of the originator or enter:

<user name>

6 In the **Priority** field enter:

<priority>

- Enter a number in the range of one (1) to ten (10).
  - One (1) has the highest priority; ten (10) has the lowest priority.
- The **Priority** field specifies the **Production Request Editor** priority to be included in the Data Processing Request(s) that result(s) from the Production Request.
- **Production Request Editor Priority** is subsequently weighted according to the value specified in the Production Strategy selected from the Planning Workbench when a Production Plan is created that uses the Production Request.

7 **Single-click** on the **PGE...** button.

- The **PGE** GUI is displayed.

8 Select the desired PGE by **single-clicking** on the PGE.

- The desired **PGE** is highlighted.
- The PGE list is scrollable. (If there are items on the list in addition to those currently visible in the window, the additional items can be viewed by clicking on the arrows associated with the scrollbars.)
- The **Find** button provides a means of performing a keyword search of the **PGE Selection** table.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **PGE Selection** table that has the search text is highlighted.)
  - **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

9 **Single-click** the appropriate button from the following selections:

- **OK -** to approve the selected PGE and dismiss the **PGE Selection** page.
  - The **Production Request - PR Edit** GUI is displayed.
  - The following fields are automatically filled in on the **Production Request - PR Edit** GUI: **Satellite Name**, **Instrument Name**, **PGE Name**, **PGE Version,** and **Profile ID.**
- **Cancel** - to return to the **Production Request - PR Edit** GUI without specifying a PGE.
  - The **Production Request - PR Edit** GUI is displayed.

10 **Single-click** on the **PGE Parameters...** button.

- The **PGE Parameters** GUI is displayed.
- PGE parameter data are displayed in a table that has the following columns:
  - **Parameter Name.**
  - **Logical Id.**
  - **Default Value.**
  - **Override Value.**
  - **Description.**

11 Select the parameter (if any) with the default value to be changed by **single-clicking** on the **Parameter Name** row.

- The parameter row is highlighted.
- Only the default values can be modified.
- Modify parameter values when and as directed by the customer (e.g., MODIS) only.

12 If a parameter with a default value to be changed has been selected, in the **Override Value** field enter:

<override value>

13 If a parameter with a default value to be changed has been selected, **single-click** on the **Apply** button at the bottom of the **PGE Parameter Mappings** window.

- The new value is displayed in the **Override Value** column.

14 If any other parameter is to be changed, repeat Steps 11 through 13.

15 **Single-click** on the appropriate button from the following selections:

- **OK -** to approve the change(s) and dismiss the **PGE Parameter Mappings** window.
  - The **Production Request - PR Edit** GUI is displayed.
- **Cancel** - to return to the **Production Request - PR Edit** GUI without saving the new value(s).
  - The **Production Request - PR Edit** GUI is displayed.

*****The procedures that follow involve the implementation of specific PRODUCTION RULES*****

MODIS *uses* 1) *Temporal Rules, to include Basic and Advanced Temporal specifications,*
2) *Orbit-Based Activation* ***and***
3) *Period/Calendar Specification,*
4) *Conditional Activation,*
5) *Additional inputs* ***and***
6) *DataBase Query.*

ASTER *uses both Temporal Rules, Basic and Advanced Temporal specifications. In addition, ASTER routine processing makes use of the Optional DPRs Production Rule to schedule and execute ASTER PGEs for new data that have been archived. The Production Planner specifies the* **insertion time** *range (i.e., the time period when the desired data were archived) as opposed to the* **collection time** *(when the satellite instrument gathered the data).*

MISR *has primarily "orbit" based PGEs.*

Multiple "production rules" can be combined to complete a PR, however, ***Temporal- and Orbit-based rules*** *cannot be combined.*

To execute either a ***Basic or Advanced Temporal Production Rule****, you must complete Steps 1 through 15, specified above in Section 13.2.3 and specify date and time information for processing (Steps 30 and 31). Then continue processing with Step 36, etc. as necessary.*

METADATA-BASED PRODUCTION RULE

16 If the PGE is subject to a metadata-based production rule and the value(s) to be checked need(s) to be changed, **single-click** on the **Metadata Checks…** button, then perform Steps 17 through 21 as applicable; otherwise go to Step 22.

- The **MetadataChecks** GUI page is displayed.
- The **MetadataChecks** GUI has an **InputDataType** window that lists the input data types for the PGE.
- In addition, the **MetadataChecks** GUI has a metadata checks (**MetaDataField-Operator-Value-Type**) window in which there is a table that lists the following information concerning each metadata check:
  - **MetaDataField.**
  - **Operator.**
  - **Value.**
  - **Type.**

- Initial values for metadata checks are entered during SSI&T; however, it is possible to modify the values using the **MetadataChecks** GUI when creating a production request.
  - Modify metadata check values when and as directed by the customer (e.g., MODIS) only.

17 If it is necessary to change any value(s) for metadata checks, select an input data type with a value to be changed by **single-clicking** on the corresponding row in the **InputDataType** window.

- The input data type row is highlighted.
- The metadata check information for the highlighted input data type is displayed in the **MetaDataField-Operator-Value-Type** window.
- The **Find** button provides a means of performing a keyword search of the **InputDataType** window.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **InputDataType** window that has the search text is highlighted.)
  - **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

18 Select (highlight) a metadata field with a comparison value to be changed by **single-clicking** on the corresponding row in the **MetaDataField-Operator-Value-Type** window.

- The metadata field row is highlighted in the **MetaDataField-Operator-Value-Type** window.
- The identity of the metadata field is displayed in the **MetaDataField** window.
- The **Find** button provides a means of performing a keyword search of the **MetaDataField-Operator-Value-Type** window.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **MetaDataField-Operator-Value-Type** window that has the search text is highlighted.)
  - **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

19 If it is necessary to change any value(s) for metadata checks, in the **Value** field enter:

<value>

20 **Single-click** on the appropriate button from the following selections:

- **OK** - to approve the new value and dismiss the **MetadataChecks** page.
  - The **Production Request - PR Edit** GUI is displayed.
  - Go to Step 22.

- **Apply** - to approve the new value without dismissing the **MetadataChecks** page.
  - Go to Step 21.
- **Cancel** - to return to the **Production Request - PR Edit** GUI without saving the new value.
  - The **Production Request - PR Edit** GUI is displayed.
  - Go to Step 22.

21 If any additional value(s) to be checked need to be changed, repeat Steps 17 through 20 as necessary.

ALTERNATE INPUTS PRODUCTION RULE

22 If the PGE is subject to the **Alternate Inputs Production Rule** and the timer settings or the order of alternate inputs need to be changed, **single-click** on the **Alternate Input Values…** button and perform Steps 23 through 28 as applicable; otherwise go to Step 29.

- The **AlternateInputValues** GUI page is displayed.
- The **AlternateInputValues** GUI has an **AlternateListName** window that lists the applicable alternate inputs.
- In addition, the **AlternateInputValues** GUI has an alternate input (**Order-DataType-LogicalID-Timer**) window in which there is a table that lists the following information concerning each alternate input:
  - **Order.**
  - **DataType.**
  - **LogicalID.**
  - **Timer.**
- The initial set-up for alternate inputs is entered during SSI&T; however, it is possible to modify the set-up using the **AlternateInputValues** GUI when creating a production request.

23 If it is necessary to change timer settings or the order of alternate inputs, first select (highlight) an alternate input to be changed by **single-clicking** on the corresponding row in the **AlternateListName** window.

- The alternate input row is highlighted.
- Information concerning the highlighted alternate input is displayed in the **Order-DataType-LogicalID-Timer** window.
- The **Find** button provides a means of performing a keyword search of the **AlternateListName** window.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **AlternateListName** window that has the search text is highlighted.)
  - **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

24 Select (highlight) an alternate input with timer settings or the order of alternate inputs to be changed by **single-clicking** on the corresponding row in the **Order-DataType-LogicalID-Timer** window.

- Alternate input row is highlighted in the **Order-DataType-LogicalID-Timer** window.
- The data type of the alternate input is displayed in the **DataType** field.

25 If it is necessary to change the order of alternate inputs, **single-click** on the up/down arrow buttons adjacent to the **Order-DataType-LogicalID-Timer** window as necessary until the highlighted alternate input has the proper order listed in the **Order** column of the window.

- If necessary, repeat Steps 24 and 25 to change the order of additional alternate inputs.

26 If the timer setting for an alternate input is to be modified, verify that the alternate input with the timer setting to be changed has been highlighted then in the **Timer** fields enter:

<hh:mm:ss>

- Another method of changing timer settings (other than typing the numbers) is to **single-click** in each of the timer fields in turn and click on the up/down buttons adjacent to the **Timer** fields until the correct time is indicated.

27 **Single-click** on the appropriate button from the following selections:

- **OK -** to approve the new alternate input setting(s) and dismiss the **AlternateInputValues** GUI.
  - The **Production Request - PR Edit** GUI is displayed.
  - Go to Step 29.
- **Apply** - to approve the new alternate input setting(s) without dismissing the **AlternateInputValues** GUI.
  - Go to Step28.
- **Cancel** - to return to the **Production Request** - **PR Edit** GUI without saving the new alternate input setting(s).
  - The P**roduction Request - PR Edit** GUI is displayed.
  - Go to Step 29.

28 If any additional alternate input setting(s) need to be changed, repeat Steps 23 through 27 as necessary.

TIME- OR ORBIT-BASED PROCESSING?

Depending on the specific PGE chosen, either the **Collection Time** or **Insertion Time** button is automatically selected. **Collection Time** (time when the data were collected by the instrument on the satellite) is used for most PGEs. The **Insertion Time** option is available primarily for ASTER processing to allow the generation of DPRs for all data contained on an ASTER tape received from the ASTER Ground Data System (GDS). If the **Insertion Time** option is selected, the **Multiple DPRs** toggle button appears on the **PR Edit** GUI.

29 **Single-click** on either the **UTC Time** (Coordinated Universal Time) button or the **Orbit** button, depending on whether data to be processed is specified by time or orbit.

- If **UTC Time** is selected, go to Step 30.
- If **Orbit** is selected go to Step 33.

TEMPORAL PRODUCTION RULES

30 Enter the desired data start date and time in the **Begin** fields in the following format:

<MM/DD/YYYY hh:mm:ss>

- As data are entered in each field the cursor automatically advances to the next field.
- Another method of entering date and time (other than typing the numbers) is to **single-click** in each of the date/time fields in turn and click on the up/down buttons adjacent to the date/time fields until the correct date/time is indicated.

31 Enter the desired data end date and time in the **End** fields in the following format:

<MM/DD/YYYY hh:mm:ss>

MULTIPLE DPRS FOR INSERTION TIME PRODUCTION RULE

32 If the Multiple DPRs for Insertion Time Production Rule applies, ensure that the **Multiple DPRs** toggle button is depressed.

- If necessary, **single-click** on the **Multiple DPRs** toggle button.
- If the Multiple DPRs for Insertion Time Production Rule does not apply, go to Step 33.

ORBITAL PROCESSING PRODUCTION RULE

33 If the Orbital Processing Production Rule applies, in the **From** field enter:

<first orbit number>

34 If the Orbital Processing Production Rule applies, in the **To** field enter:

<last orbit number>

TILING PRODUCTION RULE

35 If the Tiling Production Rule applies, in the **Tile Id** field enter:

<tile ID>

CHAIN HEAD DESIGNATION

36 **Single-click** on the appropriate button from the following selections under **PGE Chain Head**:

- **Yes -** to designate the DPRs resulting from the PR as chain heads.
  - If designated as chain heads, the DPRs produce outputs that are to be used as inputs to downstream DPRs.
- **No -** to indicate that the DPRs resulting from the PR are not chain heads.

37 If the PGE specified in the PR is at the head of a chain and should be run on a particular virtual computer, **single-click** and **hold** the button adjacent to the **Computer** field to display a menu of available virtual computers, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.

- If a particular virtual computer is selected, each DPR will be scheduled to run on the corresponding machine.
- If no virtual computer is selected, the system will try to schedule the DPRs on the machine where the bulk of the DPR's accepted inputs (both static and dynamic inputs) are already staged.

INTERMITTENT ACTIVATION

38 If Intermittent Activation applies, in the **Skip** field enter:

<number of DPRs>

- If Intermittent Activation applies, perform Steps 39 and 40.
- If Intermittent Activation does not apply, go to Step 41.

39 If Intermittent Activation applies, in the **Keep** field enter:

<number of DPRs>

40 If the first DPR is to be skipped, **single-click** on the **SkipFirst** button.

41 If applicable, in the **Comments** field enter:

<comments>

42 Start the process of saving the production request and generating DPRs by executing the following menu path:

File → Save As

- The **File Selection** window is displayed.

43 In the **Selection** field enter:

<PR name>

44 **Single-click** on the **OK** button to save the production request.

- Eventually a **Production Request Explosion into DPRs** dialogue box is displayed.
  - It may take several minutes or even hours for the process to complete.
  - If the explosion into DPRs is successful, the production request and corresponding DPR(s) are saved in the PDPS database and the Production Request **PR Name** and **Origination Date** fields are automatically updated.
  - If the explosion into DPRs is not successful, the dialogue box contains a message to that effect and it will be necessary to troubleshoot the problem.

45 **Single-click** on the **OK** button to dismiss the **Production Request Explosion into DPRs** dialogue box.

- The dialogue box is dismissed.

46 To clear the entries on the **Production Request Editor** GUI execute the following menu path:

File → New

- Return to Step 3 to create another new PR.

47 To exit the **Production Request Editor** execute the following menu path:

File → Exit

Table 13.2-4. Create a New Production Request Using the Production Request Editor GUI - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor** GUI	Use procedure in Section 13.2.2
2	**PR Edit** tab	**single-click**
3	**<PR type>** (**PR Type:** button)	**single-click**
4	**<user type>** (**User Type:** button)	**single-click**
5	**<originator>** (**Originator:** field)	**press Tab**
6	**<priority>** (**Priority:** field)	**press Tab**
7	**PGE...** button	**single-click**
8	**<PGE>** (from **PGE Selection** list)	**single-click**
9	**OK**	**single-click**
10	**PGE Parameters...** button (if applicable)	**single-click**
11	**<parameter>** (from **PGE Parameter Mappings** list) (if applicable)	**single-click**
12	**<override value>** (**Override Value** field) (if applicable)	**enter text**
13	Repeat Steps 11 and 12 (if applicable)	
14	**OK** button (if applicable)	**single-click**
15	**Metadata Checks…** button (if applicable)	**single-click**

Table 13.2-4. Create a New Production Request Using the Production Request Editor GUI - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
16	**<input data type>** (from **InputDataType** list) (if applicable)	**single-click**
17	**<metadata field>** (from **MetaDataField-Operator-Value-Type** list) (if applicable)	**single-click**
18	**<value>** (**Value** field) (if applicable)	**enter text**
19	**OK** button (if applicable)	**single-click**
20	**Alternate Input Values…** button (if applicable)	**single-click**
21	**<alternate input>** (from **AlternateListName** list) (if applicable)	**single-click**
22	**<alternate input>** (from **Order-DataType-LogicalID-Timer** list) (if applicable)	**single-click**
23	**<up>** arrow or **<down>** arrow (as necessary to reorder alternate inputs) (if applicable)	**single-click**
24	**<hh:mm:ss>** (if applicable)	**enter text**
25	**OK** button (if applicable)	**single-click**
26	Either **UTC Time** button or **Orbit** button (as applicable)	**single-click**
27	**<MM/DD/YYYY hh:mm:ss>** (**Begin** fields) (if applicable)	**enter text, press Tab**
28	**<MM/DD/YYYY hh:mm:ss>** (**End** fields) (if applicable)	**enter text, press Tab**
29	**Multiple DPRs** toggle button (if applicable)	**single-click**
30	**<first orbit number>** (**From** field) (if applicable)	**enter text**
31	Enter **<last orbit number>** (**To** field) (if applicable)	**enter text**
32	**<tile ID>** (**Tile Id** field) (if applicable)	**enter number**
33	**Yes** button (under **PGE Chain Head**) (if applicable)	**single-click**
34	**<Computer>** (**Computer** field) (if applicable)	**single-click**
35	**<number of DPRs>** (**Skip** field) (if applicable)	**enter number**
36	**<number of DPRs>** (**Keep** field) (if applicable)	**enter number**
37	**SkipFirst** button (if applicable)	**single-click**
38	**<comment>** (**Comment** field) (if applicable)	**enter text**
39	**File → Save As**	**single-click**
40	<**PR name>** (**Selection** field)	**enter text**
41	**Ok** button	**single-click**
42	**Ok** button	**single-click**

13.2.4 Create New Production Requests Using the Production Request Generator (Command-Line Interface)

The process of creating new Production Requests using the Production Request (PR) Generator (EcPlPRGenerator) begins with the Production Planner preparing an input file for the PR Generator. Then the Production Planner starts the PR Generator, referencing the input file, and the PR Generator generates the appropriate PRs.

Before creating the new PRs the Production Planner must be prepared to provide the PGE ID (PgeId) and GEO ID (GEOId) values to be used in preparing the PRs. PgeId identifies a PGE. GEOId is the portion of a granule UR that identifies granule type, subtype, and database ID in the format ***type:subtype:dbID***. For example, SC:AST_ANC.001:14754 is the GEOId for a science granule of ASTER ancillary data with a database ID of 14754.

Table 13.2-5 presents (in a condensed format) the steps required to create new production requests using the PR Generator (command-line interface). If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Access a terminal window logged in to the Planning/Management Workstation host.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 To set the application environment variables enter:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

3 In the terminal window, at the command line, enter:

cd <path>

- **<path>** is the path (e.g., **/usr/ecs/OPS/CUSTOM/data/PLS**) specified as the value for the EcPlPRGenerator's **InfoPath** configuration parameter.
  - Refer to the EcPlPRGenerator configuration parameters in the Registry database or the EcPlPRGenerator.CFG (EcPlPRGenerator.CFG.rgy) file.

4 Create an input file using an appropriate text editor (e.g., **vi**).

- For example:

  x0pls01{cmops}[10]->vi PRfile

  ~

  ~

  ~

  […]

  "PRfile" [New file]

  – Many blank lines have been deleted from the example.
- The new file will specify the PgeId and GEOId values to be used in generating new Production Requests.
- Although this procedure has been written for the **vi** editor, any UNIX editor can be used to create the file.

5 Enter:

<PgeId> <GEOId >

- For example the following entries are included in **PRfile** (the preceding example):

  ACT#syn1#001 SC:AST_L1B.001:76677
  BTS#syn1#001 SC:AST_L1B.001:76677

  – **ACT#syn1#001** and **BTS#syn1#001** are PgeId values.
  – **SC:AST_L1B.001:76677** is a GEOId.
- The following vi editor commands are useful:
  – **h** (move cursor left).
  – **j** (move cursor down).
  – **k** (move cursor up).
  – **l** (move cursor right).
  – **a** (append text).
  – **i** (insert text).
  – **r** (replace single character).
  – **x** (delete a character).
  – **dw** (delete a word).
  – **dd** (delete a line).
  – ***n*dd** (delete *n* lines).
  – **u** (undo previous change).
  – **Esc** (switch to command mode).

6 Repeat Step 5 for each production request to be created.

7 If using **vi**, when all appropriate entries have been made in the file press the **Esc** key.

8 Save the file.

- For example, if using **vi**, enter:

  ZZ

  – **vi** exits and the new file is saved.
  – To exit **vi** without saving the new entries in the file enter:

    :q!

- UNIX prompt is displayed.

9 In the terminal window, at the command line, enter:

cd ../utilities

- Change directory to the directory containing the production planning startup scripts (e.g., EcPlPRE_IFStart).

10 To start the PR Generator enter:
EcPlPRGeneratorStart <MODE> <FileName>

11 When the PR Generator has completed its operation, enter:

cd ../logs

- Change directory to the logs directory.

12 In the terminal window, at the command line, enter:

pg EcPlPRGeneratorDebug.log

- The first page of the PR Generator debug log is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **view, vi**) can be used to review the log file.

13 Review the log file to determine the results of running the PR Generator.

- For example (extract from EcPlPRGeneratorDebug.log):

  *********** THE STATUS OF PRODUCTIONREQUESTS: ************

  ACT#syn1#001 SC:AST_L1B.001:76677
  ******** The above enties in Input file Failed because of bad GEOID's********

- To exit from **pg** at the **:** prompt enter:

  q

  – The command line prompt is displayed.

Table 13.2-5. Create New Production Requests Using the Production Request Generator (Command-Line Interface) - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**setenv ECS_HOME /usr/ecs/**	**enter text, press Enter**
3	**cd <path>**	**enter text, press Enter**
4	Create an input file using an appropriate text editor (e.g., **vi <file name>**)	**enter text, press Enter**
5	**<PgeId> <GEOId >**	**enter text, press Enter**
6	Repeat Step 5 (as necessary)	
7	**Esc** key (if/when appropriate)	**press**
8	Save the file (e.g., **ZZ**)	**enter text, press Enter**
9	**cd ../utilities**	**enter text, press Enter**
10	**EcPlPRGeneratorStart <MODE> <FileName>**	**enter text, press Enter**
11	**cd ../logs** (when appropriate)	**enter text, press Enter**
12	**pg EcPlPRGeneratorDebug.log**	**enter text, press Enter**
13	Review the log file to determine the results of running the PR Generator	**read text**

13.2.5 Edit/Modify a Production Request

Edits or modifications to a previously created production request result in a new production request. The new production request must be saved with a new name.

Table 13.2-6 presents (in a condensed format) the steps required to edit/modify a production request. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Launch the **Production Request Editor** (refer to Section 13.2.2).

- The **Production Request Editor** GUI is displayed.

2 **Single-click** on the **PR Edit** tab.

- The **PR Edit GUI** is displayed.

3 To display a list of Production Requests from which to select the PR to be opened execute the following menu path:

File → Open

- A list of Production Requests is displayed in the **File Selection** window.

4 Select (highlight) the PR to be edited/modified by **single-clicking** on the corresponding PR name in the list of PRs.

5 **Single-click** on the **OK** button.

- The PR information appears in the **PR Edit GUI**.

NOTE: Perform only those steps of the procedure that are applicable to the changes you want to make. You do not have to go through all of the fields in the PR to successfully modify it. However, you must save the modified PR to make the changes effective.

6 To change the type of production request select the desired **PR Type** from the option button by **single-clicking** and **holding**, **moving** the mouse cursor to the desired selection (highlighting it), then **releasing** the mouse button.

7 To change the user type select the desired **User Type** from the option button by **single-clicking** and **holding**, **moving** the mouse cursor to the desired selection (highlighting it), then **releasing** the mouse button.

8 To change the originator in the **Originator** field either enter the actual name of the originator or enter:

<user name>

9 To change the **Production Request Editor Priority** in the **Priority:** field enter:

<priority>

10 If changing the PGE, **single-click** on the **PGE...** button.

- The **PGE** GUI is displayed.

11 If changing the PGE, select the desired PGE by **single-clicking** on the PGE.

12 If changing the PGE, **single-click** the **Ok** button.

13 To change PGE parameters first **single-click** on the **PGE Parameters...** button.

- The **PGE Parameters** GUI displays.

14 When changing PGE parameters, select a parameter to be modified by **single-clicking** on the **Parameter Name** row.

- The **parameter** row is highlighted.

- The **Find** button provides a means of performing a keyword search of the PGE Parameters table.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the PGE Parameters table that has the search text is highlighted.)
  - **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

15 When changing PGE parameters, in the **Override Value** field enter:

<override value>

- Some PGEs may NOT have modifiable parameters.

16 When changing PGE parameters, **single-click** on the **Apply** button at the bottom of the **PGE Parameter Mappings** window.

17 When changing PGE parameters, **single-click** on the **Ok** button to approve the changes.

- The **Production Request Editor** GUI is displayed.

18 If the PGE is subject to a metadata-based production rule and the value(s) to be checked need(s) to be changed, **single-click** on the **Metadata Checks…** button.

19 If it is necessary to change any value(s) for metadata checks, select an input data type with a value to be changed by **single-clicking** on the corresponding row in the **InputDataType** window.

20 If it is necessary to change any value(s) for metadata checks, select (highlight) a metadata field with a comparison value to be changed by **single-clicking** on the corresponding row in the **MetaDataField-Operator-Value-Type** window.

21 If it is necessary to change any value(s) for metadata checks, in the **Value** field enter:

<value>

22 If it is necessary to change any value(s) for metadata checks, **single-click** on the appropriate button from the following selections:

- **OK** - to approve the new value and dismiss the **MetadataChecks** GUI.
- **Apply** - to approve the new value without dismissing the **MetadataChecks** GUI.
- **Cancel** - to return to the **Production Request** - **PR Edit** GUI without saving the new value.

23 If the PGE is subject to the **Alternate Inputs Production Rule** and the timer settings or the order of alternate inputs need to be changed, **single-click** on the **Alternate Input Values…** button.

24 If it is necessary to change timer settings or the order of alternate inputs, first select (highlight) an alternate input to be changed by **single-clicking** on the corresponding row in the **AlternateListName** window.

25 If it is necessary to change timer settings or the order of alternate inputs, select (highlight) an alternate input with timer settings or the order of alternate inputs to be changed by **single-clicking** on the corresponding row in the **Order-DataType-LogicalID-Timer** window.

26 If it is necessary to change the order of alternate inputs, **single-click** on the up/down arrow buttons adjacent to the **Order-DataType-LogicalID-Timer** window as necessary until the highlighted alternate input has the proper order listed in the **Order** column of the window.

27 If the timer setting for an alternate input is to be modified, verify that the alternate input with the timer setting to be changed has been highlighted then in the **Timer** fields enter:

<hh:mm:ss>

28 If it is necessary to change timer settings or the order of alternate inputs, **single-click** on the appropriate button from the following selections:

- **OK** - to approve the new alternate input setting(s) and dismiss the **AlternateInputValues** GUI.
- **Apply** - to approve the new alternate input setting(s) without dismissing the **AlternateInputValues** GUI.
- **Cancel** - to return to the **Production Request** - **PR Edit** GUI without saving the new alternate input setting(s).

29 **Single-click** on either the **UTC Time** (Coordinated Universal Time) button or the **Orbit** button, depending on whether data to be processed is specified by time or orbit.

NOTE: Depending on the specific PGE chosen, either the **Collection Time** or **Insertion Time** button is automatically selected. If the **Insertion Time** option is selected, the **Multiple DPRs** toggle button appears on the **PR Edit** GUI.

30 If **UTC Time** is to be changed, enter the desired data start date and time in the **Begin** fields in the following format:

<MM/DD/YYYY hh:mm:ss>

31 If **UTC Time** is to be changed, enter the desired data end date and time in the **End** fields in the following format:

<MM/DD/YYYY hh:mm:ss>

32 If the Multiple DPRs for Insertion Time Production Rule applies, ensure that the **Multiple DPRs** toggle button is depressed.

- If necessary, **single-click** on the **Multiple DPRs** toggle button.

33 If the Orbital Processing Production Rule applies and the first orbit of data is to be modified, in the **From** field enter:

<first orbit number>

34 If the Orbital Processing Production Rule applies and the last orbit of data is to be modified, in the **To** field enter:

<last orbit number>

35 If the Tiling Production Rule applies, in the **Tile Id** field enter:

<tile ID>

36 **Single-click** on the appropriate button from the following selections under **PGE Chain Head**:

- **Yes -** to designate the DPRs resulting from the PR as chain heads.
- **No -** to indicate that the DPRs resulting from the PR are not chain heads.

37 If the PGE specified in the PR is at the head of a chain and should be run on a particular virtual computer, **single-click** and **hold** the button adjacent to the **Computer** field to display a menu of available virtual computers, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.

38 If Intermittent Activation applies and the number of DPRs to skip is to be modified, in the **Skip** field enter:

<number of DPRs>

39 If Intermittent Activation applies and the number of DPRs to keep is to be modified, in the **Keep** field enter:

<number of DPRs>

40 If Intermittent Activation applies and the first DPR is to be skipped, verify that the **SkipFirst** button has been activated (**single-click** on the button if necessary).

41 If applicable, in the **Comments** field enter:

<comments>

42 To save the modified production request execute the following menu path:

File → Save As

- **File Selection** Window appears.

43 In the **Selection** field enter:

<PR name>

- Production Request is named.
- Name must be changed.

44 **Single-click** on the **Ok** button.

- Eventually a **Production Request Explosion into DPRs** dialogue box is displayed.

45 **Single-click** on the **OK** button to dismiss the **Production Request Explosion into DPRs** dialogue box.

- The dialogue box is dismissed.

Table 13.2-6. Edit/Modify a Production Request - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor**	Use procedure in Section 13.2.2
2	**PR Edit** tab	**single-click**
3	**File → Open**	**single-click**
4	**<PR name>** (from **Production Requests:** list)	**single-click**
5	**OK** button	**single-click**
6	**<PR type>** (**PR Type:** button) (if applicable)	**single-click**
7	**<user type>** (**User Type:** button) (if applicable)	**single-click**
8	**<originator>** (**Originator:** field) (if applicable)	**enter text**
9	**<priority>** (**Priority:** field) (if applicable)	**enter number**
10	**PGE...** button (if applicable)	**single-click**
11	**<PGE>** (from **PGE Selection** list) (if applicable)	**single-click**
12	**OK** button (if applicable)	**single-click**
13	**PGE Parameters...** button (if applicable)	**single-click**
14	**<parameter>** (from **PGE Parameter Mappings** list) (if applicable)	**single-click**

Table 13.2-6. Edit/Modify a Production Request - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
15	**<override value>** (**Override Value** field) (if applicable)	**enter text**
16	**OK** button (if applicable)	**single-click**
17	**Metadata Checks…** button (if applicable)	**single-click**
18	**<input data type>** (from **InputDataType** list) (if applicable)	**single-click**
19	**<metadata field>** (from **MetaDataField-Operator-Value-Type** list) (if applicable)	**single-click**
20	**<value>** (**Value** field) (if applicable)	**enter text**
21	**OK** button (if applicable)	**single-click**
22	**Alternate Input Values…** button (if applicable)	**single-click**
23	**<alternate input>** (from **AlternateListName** list) (if applicable)	**single-click**
24	**<alternate input>** (from **Order-DataType-LogicalID-Timer** list) (if applicable)	**single-click**
25	**<up>** arrow or **<down>** arrow (as necessary to reorder alternate inputs) (if applicable)	**single-click**
26	**<hh:mm:ss>** (**Timer** fields) (if applicable)	**enter text**
27	**OK** button (if applicable)	**single-click**
28	Either **UTC Time** button or **Orbit** button (as applicable)	**single-click**
29	**<MM/DD/YYYY hh:mm:ss>** (**Begin** fields) (if applicable)	**enter text, press Tab**
30	**<MM/DD/YYYY hh:mm:ss>** (**End** fields) (if applicable)	**enter text, press Tab**
31	**Multiple DPRs** toggle button (if applicable)	**single-click**
32	**<first orbit number>** (**From** field) (if applicable)	**enter number**
33	**<last orbit number>** (**To** field) (if applicable)	**enter number**
34	**<tile ID>** (**Tile Id** field) (if applicable)	**enter number**
35	**Yes** button (under **PGE Chain Head**:) (if applicable)	**single-click**
36	**<Computer>** (from **Computer** field) (if applicable)	**single-click**
37	**<number of DPRs>** (**Skip** field) (if applicable)	**enter number**
38	**<number of DPRs>** (**Keep** field) (if applicable)	**enter number**
39	**SkipFirst** button (if applicable)	**single-click**
40	**<comment>** (**Comment** field) (if applicable)	**enter text**
41	**File → Save As**	**single-click**
42	**<PR name>** (**Selection** field)	**enter text**
43	**Ok** button	**single-click**
44	**Ok** button	**single-click**

13.2.6 Delete a Production Request

Production Requests can be deleted if necessary. Table 13.2-7 presents (in a condensed format) the steps required to delete Production Requests. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Launch the **Production Request Editor** (refer to Section 13.2.2).

- The **Production Request Editor** GUI is displayed.

2 **Single-click** on the **PR List** tab.

- The **PR List** GUI is presented.
- A list of Production Requests is displayed.

3 If filtering of the list of Production Requests is desired, **single-click** and **hold** the **PR Type** option button to display a menu of types of Production Requests, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.

- Options are: **All** [default selection], **Routine**, **OnDemand**, and **Reprocessing**.
- A list of Production Requests of the specified type is displayed.
- A **Refresh** button is available for refreshing the list if necessary.
- The **Find** button provides a means of performing a keyword search of the **PR List** GUI page.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **PR List** GUI page that has the search text is highlighted.)
  - **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

4 **Single-click** on the Production Request to be deleted.

- The Production Request to be deleted is highlighted.

5 Execute the following menu path:

Edit→Delete

- A dialogue box is displayed requesting confirmation of the decision to delete the Production Request.

6 **Single-click** on the **OK** button to delete the Production Request.

- If there are no dependencies among Production Requests, a confirmation dialogue box is displayed.
- If there are dependencies among Production Requests, a “List Of Orphan DPRs” dialogue box is displayed.

- The **Find** button provides a means of performing a keyword search of the “List Of Orphan DPRs” dialogue box.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the “List Of Orphan DPRs” dialogue box that has the search text is highlighted.)
  - **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

7 **Single-click** on the appropriate button from the following selections:
- **OK** - to confirm deletion of the Production Request(s) and dismiss the dialogue box.
  - A “deletion completed” dialogue box is displayed.
  - The Production Request(s) is/are deleted.
- **Cancel** - to preserve the Production Request(s) and dismiss the dialogue box.
  - The dialogue box is dismissed.
  - The Production Request(s) is/are not deleted.

8 **Single-click** on **OK** to dismiss the “deletion completed” dialogue box.
- The “deletion completed” dialogue box is dismissed.

9 To start the process of exiting from the **Production Request Editor** GUI execute the following menu path:

File → Exit

- A **Do you really want to exit?** dialogue box is displayed.

10 **Single-click** on the appropriate button from the following selections:
- **OK** - to exit from the Production Request Editor GUI.
  - The **Production Request Editor** GUI is dismissed.
- **Cancel** - to return to the **Production Request Editor** GUI.

Table 13.2-7. Delete a Production Request - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor**	Use procedure in Section 13.2.2
2	**PR List** tab	**single-click**
3	**<PR name>** (from **Production Requests** list)	**single-click**
4	**Edit→Delete**	**single-click**
5	**OK** button	**single-click**
6	**OK** button	**single-click**
7	**OK** button	**single-click**

13.2.7 Review Data Processing Requests

The process of reviewing Data Processing Requests begins with the Production Planner launching the Production Request Editor. The Production Planner can review DPRs associated with a specific PR. The Production Planner can review such DPR values as input granule(s), output granule(s), predicted and actual start times, data start time, status, and priority.

Table 13.2-8 presents (in a condensed format) the steps required to display and review DPRs. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Launch the **Production Request Editor** (refer to Section 13.2.2).

- The **Production Request Editor** GUI is displayed.

2 **Single-click** on the **DPR List** tab.

- The **DPR List** GUI is presented.

3 If filtering of the list of Production Requests is desired, **single-click** and **hold** the **PR Type** option button to display a menu of types of Production Requests, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.

- Options are: **ALL** [default selection], **Routine**, **OnDemand**, and **Reprocessing**.

4 **Single-click** on the option button associated with the **PR Name** field.

- A list of Production Requests is displayed.

5 **Single-click** on the Production Request that includes the DPR to be reviewed.

- The Production Request is entered into the **PR Name** field.

6 **Single-click** on the **Filter** button.

- The DPRs for the specified PR only are shown in the **Data Processing Requests** table.

7 **Single-click** on the DPR to be reviewed from those shown in the **Data Processing Requests** table.

- The DPR to be reviewed is highlighted.
- The **Find** button provides a means of performing a keyword search of the **Data Processing Requests** table.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **Data Processing Requests** table that has the search text is highlighted.)

- **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

8 Execute the following menu path:

File → Open

9 **Single-click** on the **DPR View** tab.

- The **DPR View** GUI is displayed.
- The **DPR** information is displayed on the **DPR View** GUI.

10 Review **Data Processing Request Identification** information displayed.

- **DPR Name.**
- **PR Name.**
- **Origination Date**
- **Originator**
- **PGE Id.**
- **Data Start Time.**
- **Data Stop Time.**
- **Predicted Start Time (date and time).**
- **Actual Start Time (date and time).**
- **Priority.**
- **Status.**

11 **Single-click** on the **PGE Parameters...** button.

- The **PGE Parameter Mappings** GUI displays.

12 Review the **PGE Parameter Mappings** information displayed.

- PGE parameter data are displayed in a table that has the following columns:
  - **Parameter Name.**
  - **Logical Id.**
  - **Default Value.**
  - **Override Value.**
  - **Description.**
- The **Find** button provides a means of performing a keyword search of the PGE Parameters table.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the PGE Parameters table that has the search text is highlighted.)
  - **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

13 When finished reviewing the PGE parameters, **single-click** on the **OK** button.

- The **PGE Parameter Mappings** GUI is dismissed.
- The **DPR View** GUI is displayed.

14 **Single-click** on the **PGE File Mappings...** button.

- The **UR File Mappings** GUI is displayed.

15 Review the **Universal Reference (UR) File Mappings** information displayed.

- **Input Data.**
  - **Logical Id.**
  - **Granule Id.**
  - **Start Time(UTC)** (date and time).
  - **Stop Time (UTC)** (date and time).
  - **Availability** (date and time).
  - **UR** (granule universal reference).
- **Output Data** (Displays the same data as shown for Input Data).
- The **Find** buttons provides a means of performing keyword searches of the **Input Data** and **Output Data** tables.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. [The first entry in the **Input Data** or **Output Data** table (as applicable) that has the search text is highlighted.]
  - **Single-click** on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

16 When finished reviewing the **UR File Mappings**, **single-click** on the **OK** button.

- The **UR File Mappings** GUI is dismissed.
- The **DPR View** GUI is displayed.

17 Review the **DPR Status** information displayed.

- **Predicted Start Time** (date and time).
- **Actual Start Time** (date and time).
- **Priority.**
- **Status.**
- **Predicted** and **Actual Start Times** and **Status** are not displayed if the Production Request has not been scheduled.

18 Repeat Steps 2 through 17 to review multiple DPRs associated with multiple PRs.

19 To start the process of exiting from the **Production Request Editor** GUI execute the following menu path:

File → Exit

- A **Do you really want to exit?** dialogue box is displayed.

20 **Single-click** on the appropriate button from the following selections:
- **OK** - to exit from the **Production Request Editor** GUI.
  – The **Production Request Editor** GUI is dismissed.
- **Cancel** - to return to the **Production Request Editor** GUI.

Table 13.2-8. Display Data Processing Request Information - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor**	Use procedure in Section 13.2.2
2	**DPR List** tab	**single-click**
3	**<PR name>** (from **PR Name** option list)	**single-click**
4	**Filter** button	**single-click**
5	**<DPR ID>** (from **Data Processing Requests** list)	**single-click**
6	**File → Open**	**single-click**
7	**DPR View** tab	**single-click**
8	Review DPR information	**read text**
9	**PGE Parameters...** button	**single-click**
10	Review PGE parameters	**read text**
11	**OK** button	**single-click**
12	**PGE File Mappings...** button	**single-click**
13	Review input and output granule information	**read text**
14	**OK** button	**single-click**
15	Repeat Steps 2 through 14 to review additional DPRs	

13.2.8 Delete a Data Processing Request

DPRs can be deleted manually if necessary. Table 13.2-9 presents (in a condensed format) the steps required to delete DPRs. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Launch the **Production Request Editor** (refer to Section 13.2.2).
- The **Production Request Editor** GUI is displayed.

2 **Single-click** on the **DPR List** tab.
- The **DPR List** GUI is presented.

3 If filtering of the list of Production Requests is desired, **single-click** and **hold** the **PR Type** option button to display a menu of types of Production Requests, **move** the mouse cursor to the desired selection (highlighting it), then **release** the mouse button.

- Options are: **ALL** [default selection], **Routine**, **OnDemand**, and **Reprocessing**.

4 **Single-click** on the **PR Name** option button.

- A list of Production Requests is displayed.

5 **Single-click** on the Production Request for which a DPR listing is desired.

- The Production Request is entered into the **PR Name** field.

6 **Single-click** on the **Filter** button.

- A list of the DPRs associated with the Production Request is displayed.

7 **Single-click** on the DPR to be deleted.

- The DPR to be deleted is highlighted.
- The **Find** button provides a means of performing a keyword search of the **Data Processing Requests** table.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **Data Processing Requests** table that has the search text is highlighted.)
  - Click on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

8 Execute the following menu path:

Edit→Delete

- A DPR deletion confirmation dialogue box is displayed requesting confirmation of the decision to delete the DPR.

9 Click on the appropriate button from the following selections:

- **OK** - to confirm deletion of the DPR and dismiss the dialogue box.
  - A deletion dialogue box is displayed.
  - The DPR is deleted.
  - If the DPR was the only DPR in the Production Request, the PR is deleted too.
- **Cancel** - to preserve the DPR and dismiss the dialogue box.
  - The deletion confirmation dialogue box is dismissed.
  - The DPR is not deleted.

10 Select **OK** to dismiss the dialogue box.

- The dialogue box is dismissed.

11 To start the process of exiting from the **Production Request Editor** GUI execute the following menu path:

File → Exit

- A **Do you really want to exit?** dialogue box is displayed.

12 **Single-click** on the appropriate button from the following selections:

- **OK** - to exit from the **Production Request Editor** GUI.
  - The **Production Request Editor** GUI is dismissed.
- **Cancel** - to return to the **Production Request Editor** GUI.

Table 13.2-9. Delete a Data Processing Request - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Request Editor**	Use procedure in Section 13.2.2
2	**DPR List** tab	**single-click**
3	**<PR name>** (from **PR Name** option list)	**single-click**
4	**Filter** button	**single-click**
5	**<DPR ID>** (DPR to be deleted) (from **Data Processing Requests** list)	**single-click**
6	**Edit → Delete**	**single-click**
7	**OK** button	**single-click**
8	**OK** button	**single-click**

13.2.9 Re-Generate Granules Affected by Loss of Files from the Archive

The entry point for this procedure is the "Data Recovery Process" in Chapter 17, including (especially) the "Data Recovery Procedure for Known Files" and "SDSRV Retrieval of Granule Production History Metadata" sections.

The role of this procedure is to initiate production of replacements for granules previously generated within this DAAC but now lost due to failure in the ECS Archive. In order to achieve this, Production Requests (PRs) for the generation of the replacement granules must be created, entered into a Production Plan, and activated.

The PRs are created using the Production Request Editor (PRE). However, first the necessary parameters of the PRs must be retrieved from the Production History file (PH) of the lost granule. The PH file is itself acquired by use of the QA Monitor tool and, though the PH UR is supplied in the input to this process, this can be achieved only by first using the QA Monitor tool to search the SDSRV database for granules matching the lost granule. Hence, this process commences with use in the QA Monitor GUI of the attributes of each lost granule (shortname, versionID etc.) supplied in the input to this procedure. Refer to Chapter 17 for further information on the wider context of this procedure.

The input to this procedure is the "Granules for PDPS Re-Generation" list. This is a list of granules and associated metadata generated by the procedure "SDSRV Retrieval of Granule Production History Metadata" (refer to Chapter 17). While the list can be obtained electronically, its use in this procedure is line-by-line. However, if available electronically, entries from it can be 'cut and pasted' into the input fields of the QA monitor GUI to avoid typing errors.

The outputs of the procedure are as follows:

- Granule re-generation Production Requests entered into a production plan.
- The "PDPS Residual Granules List" which is a list of Granules which this PDPS can not re-generate or which it has been decided do not justify re-generation. This list is returned to the process in Chapter 17.

Note that the following assumptions apply to the application of this procedure:

- The Science Data Server (SDSRV) will provide a list of granules to be regenerated ("Granules for PDPS Re-Generation"). This list will contain information about the granules to be regenerated and a Universal Reference (UR) for the associated Production History tar file.
- When reproducing lost granules, all outputs of the PGE, not just those equivalent to the lost granule(s), will be produced and archived.
- It cannot be guaranteed that the PGE re-run will use the same inputs as were used during its original execution due to the variability of: Optional/Alternate inputs, Metadata Checks, Metadata Query and other Production Rules.
- It is possible that at the time of the original run of PGE, certain optional/alternate inputs were not available, which became available later. During the re-run of the PGE use of those additional or other optional inputs cannot be avoided. However, it can be assumed that an equivalent or better product than the original will be produced as a result.
- PDPS maintains a minimal amount of granule level versioning. By design, only the latest version of the granule will be used. If the PGE which is to be re-run uses inputs which have more than one granule level version, PDPS will use only the latest version of those inputs. However, if references to those granules have been deleted from the PDPS database (a delete script, which runs periodically, cleans up unused database entries), then PDPS will choose the first one returned from SDSRV. SDSRV does not guarantee any sort of ordering in this case but PDPS will select the latest granule from those returned (Note: depends on fix to NCR ECSed16326).
- At Production Request time, the default values for metadata checks can be overridden. The new values used are stored in the PDPS database but not in the Production History. If, at the time of re-run of the PGE, the references to that PGE have been deleted from PDPS database, the default metadata checks will be used. It is possible, therefore, that these default values will cause this DPR not to be run in this instance; e.g. if the metadata checks had been changed in the original run to be less restrictive. If these types of changes to metadata checks are required in order to

get DPRs to run, then it is assumed that these defaults are saved as part of the PGE profile.

- In a manner that is similar to Metadata Checks, a Metadata Query specifies a query to be used to determine the input granules to be used for a DPR. For reasons of production timing or updated QA values, a product reproduced at a later date could have different granules used as input. Again, it can be assumed that in this case a better product will result.
- Other production rules (e.g., spatial query) could make it impossible to reproduce identical granules.
- If a PGE (PGE name, version and profile) has to support lost granules regeneration, then that PGE should not be deleted from the PDPS database. This means, in the SSIT Operational Metadata GUI, the delete flag for that PGE should not be checked.

Table 13.2-10 presents (in a condensed format) the steps required to re-generate granules affected by loss of files from the archive. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Determine whether all the granules in the input list “Granules for PDPS Re-Generation” should be reproduced.

- It is possible that some granules need not be reproduced, e.g. because a newer version of that product is available.
- Granules that need not be reproduced should be added to the “PDPS Residual Granules” list. Also, at any time during this process, if it is determined that some granules cannot be regenerated or need not be regenerated, then those granules should be included in the "PDPS Residual Granules” list.

2 Retrieve (using the QA Monitor GUI) the Production History tar file from the archive for each granule in the “Granules for PDPS Re-Generation” list that needs to be reproduced.

- Procedures for using the QA Monitor GUI are presented in Chapter 15.
- Use the lost granule’s datatype, and “begin date” and “end date” values that encompass its RangeBeginningDateTime and RangeEndingDateTime. Note that the GUI interprets the dates in UTC format.
- For each granule that meets the query conditions and is displayed on the QA Monitor GUI, the granule’s UR, its Production History tar file’s UR, and the name of the Production History tar file are shown. For only one of the granules will the URs (both the granule UR and the Production History UR) match the URs for this granule in the input list.
  - If the Query failed or did not return any hit that matched, add the granule to “PDPS Residual Granules” list.

- Retrieve the matching granule.
  - The Production History tar file will be acquired to a directory that is configurable (the name of the configuration parameter is DpPrQA_DATA_DIR, and the default value for this parameter is $ECS_HOME/<MODE>/CUSTOM/data/DPS).
  - The name of the tar file is the one that appears under the column **Production History File Name** on the QA Monitor GUI.
- Note that if more than one granule in the input list maps to the same Production History tar file, then the Production History tar file need not be retrieved multiple times.
- The Production History tar file contains the Process Control File (PCF) which has all the information needed to re-run the PGE. The following particulars have to be extracted from the PCF:
  - PGE Name.
  - PGE Version.
  - PGE Profile ID.
  - DPR Start time.
  - DPR Stop time.
  - PGE runtime parameters and their associated values.
- Identification of information in the PCF:
  - The PGE Name, PGE Version, and the PGE Profile appear in the System Runtime Parameters section of the PCF. They are concatenated (with a # sign to separate them) and appear in the place reserved for "Software ID".
  - DPR Start time appears in the User Defined Parameter Section of the PCF under the logical ID 10258.
  - DPR Stop time also appears in the User Defined Parameter Section of the PCF under logical ID 10259.
  - All other logical IDs in the User Defined Parameter Section of the PCF form the run time parameters and their associated values. Note the logical ID and its corresponding values.
  - An automated script could be written to extract the values from the PCF.

3 If the PGE name, version and profile that is extracted from the PCF does not appear as an Existing/New PGE, then add the granule that is to be regenerated to the "PDPS Residual Granules" list.

4 From the SSIT host, launch the **SSIT Manager** GUI and invoke the **PDPS Operational Metadata** GUI.

- Procedures for using SSI&T tools are presented in Chapter 11.

5 If the PGE is not registered, register the PGE using the **PDPS Science Update Metadata Update** from the **SSIT Manager** GUI.

- The PGE must be registered before a production request can be entered.

6 If it is decided not to re-register the PGE, add the granule that is to be regenerated to the "PDPS Residual Granules" list.

7 From the Planning Workstation, launch the **Production Request Editor** GUI as described in Section 13.2.2.

8 Enter a Production Request (as described in Section 13.2.3) for the relevant PGE/version/profile ID.

- Use **Reprocessing** for the **Processing Type**.
- Use the **DPR Start** and **Stop Time** listed in the Production History for the **Begin** and **End** times.
- View the default PGE runtime parameters and compare them against the runtime parameters obtained from the Production History tar file.
  - Modify the runtime parameter values to match exactly what was used in the original run.
- If granules that need to be regenerated are produced by PGEs that are chained, then the production requests must be entered in that order.
  - For instance, if granules A and B are to regenerated, and PGEs P1 & P2 produce them and if P1 & P2 are chained (P2 takes P1's outputs as its inputs) then the production request for P1 must be entered before entering one for P2.

9 Launch the **Planning Workbench** (refer to Section 13.3.1).

- The **Planning Workbench** is displayed.

10 Create and activate a production plan that includes the newly created production request(s) as described in Section 13.3.2.

11 Return the output ("PDPS Residual Granules") list to the "Data Recovery Procedure for Known Files" procedure in Chapter 17 for further processing.

- Those granules which were not re-generated by this process are listed in the output ("PDPS Residual Granules") list.

Table 13.2-10. Re-Generate Granules Affected by Loss of Files from the Archive - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Identify granules to be reproduced	**read text**
2	Add not-to-be-reproduced granules to the "PDPS Residual Granules "List	**enter text**
3	Retrieve Production History tar file for each element to be reproduced using the **QA Monitor GUI**	Use QA Monitor procedures (Chapter 15)

Table 13.2-10. Re-Generate Granules Affected by Loss of Files from the Archive - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
4	Obtain URs from the QA Monitor GUI	Use QA Monitor procedures (Chapter 15)
5	Extract PGE parameters from the Process Control File	Use QA Monitor procedures (Chapter 15)
6	Launch the **PDPS Operational Metadata GUI** from the **SSIT Manager GUI** if the applicable PGE is not registered	Use SSIT procedure for launching the PDPS Operational Metadata GUI (Chapter 11)
7	Register the PGE	Use SSIT procedure for registering PGEs (Chapter 11)
8	Launch the **Production Request Editor GUI**	Use procedure in Section 13.2.2
9	Enter the **Production Request** for **Reprocessing**	Use procedure in Section 13.2.3
10	Launch the **Planning Workbench**	Use procedure in Section 13.3.1
11	Create and activate a plan for the newly created **Production Request**	Use procedure in Section 13.3.2
12	Return the output (“PDPS Residual Granules”) list to the “Data Recovery Procedure for Known Files” procedure (Chapter 17)	

13.3 Creating/Modifying/Deleting Production Strategies

A Production Strategy is a high-level set of priorities by which the Production Planner defines the rules for priorities and preferences in the processing of DPRs. The Production Planner uses the **Production Strategies** GUI to prepare Production Strategies.

Production Strategies work on two levels. First, the Production Planner can update lists of DPR attributes so that each value an attribute can have is tied to a particular priority. For example, the DPR attribute “PR Type” has three values that may have their default priority of (e.g., 2) changed as follows (on a scale of 1 to 10):

- Routine 6
- On-Demand 10
- Reprocessing 4

Next, the Production Planner can change the weight that each attribute's priority is given. For example, weights (from 1 to 100) might be assigned to the DPR attributes as follows:

- PR Type 45
- User Type 15
- PGE Type 20

A weight is also given to the priority specified when the Production Planner created the Production Request as shown in the following example:

- Production Request Editor 20

The total weights assigned to PR Type, User Type, PGE Type and Production Request Editor [Priority] must equal 100.

The values included in the selected strategy are read by the Planning Workbench when prioritizing the DPRs in a production plan.

Finally, there is a Late Start Delta that can be used to increase the priority of all jobs that have been waiting in the Production Queue for more than a day.

Table 13.3-1 provides an Activity Checklist for activities related to the creation of Production Strategies.

Table 13.3-1. Production Strategies - Activity Checklist

Order	Role	Task	Section	Complete?
1	Production Planner	Launch the Production Strategies GUI	(P) 13.3.1	
2	Production Planner	Define or Modify a Production Strategy	(P) 13.3.2	
3	Production Planner	Review the Current Active Strategy	(P) 13.3.3	
4	Production Planner	Delete a Production Strategy	(P) 13.3.4	

13.3.1 Launch the Production Strategies GUI

The Production Strategies GUI is invoked as described in the procedure that follows. Table 13.3-2 presents (in a condensed format) the steps required to launch the Production Strategies GUI. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Access a terminal window logged in to the Planning/Management Workstation host.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 In the terminal window, at the command line, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1 - Science Software Integration and Test (SSI&T)
  - TS2 - New Version Checkout
  - OPS - Normal Operations
- “utilities” is the directory containing the Planning Subsystem start-up scripts.

3 Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

4 Start the Production Strategies GUI by entering:

EcPlProdStratStart <MODE>

- The **Production Strategies** GUI is launched.

Table 13.3-2. Launch the Production Strategies GUI - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	Set the environment variables if necessary	**enter text, press Enter**
4	**EcPlProdStratStart <MODE>**	**enter text, press Enter**

13.3.2 Define or Modify a Production Strategy

The Production Planner uses the Production Strategies GUI to develop Production Strategies. Table 13.3-3 presents (in a condensed format) the steps required to define or modify a production strategy. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Launch the **Production Strategies** GUI (refer to Section 13.3.1).

- The **Production Strategies** GUI is displayed.

2 If defining a new production strategy, execute the following menu path:

File → New

- The fields of the **Production Strategies** GUI are reset.

3 If modifying an existing production strategy, first **single-click** the option button associated with the **Production Strategies** field, then highlight (in the option menu) the name of the production strategy to be modified.

- Data pertaining to the selected production strategy are displayed in the applicable fields of the **Production Strategies** GUI.

- Alternatively, it is possible to execute the following menu path: **File** → **Open**, select the desired production strategy from the list on the **Open** window, and **single-click** on the **Ok** button to open the production strategy.

NOTE: The **Help** button at the bottom of **Production Strategies Open** window is non-functional (does not work).

4 If changing the default priority for PR Type, in the **Default** field below the **PR Type** button enter:

<default value>

- The range for the default is from 1 to 10.

5 If changing the default priority for User Type, in the **Default** field below the **User Type** button enter:

<default value>

6 If changing the default priority for PGE Type, in the **Default** field below the **PGE Type** button enter:

<default value>

7 If defining or modifying a priority for a type of production request, first **single-click** on the **PR Type** button.

- The different types of production requests are displayed in the **Type List** field at the bottom left of the GUI.
- Options are: **Routine**, **OnDemand**, and **Reprocessing**.

8 If defining a priority for a type of production request **not** currently listed in the **PR Type Value-Priority** list, **single-click** on that PR type in the **Type List** field.

- The PR type is highlighted.
- It is possible to highlight multiple PR types (by **single-clicking** on each one in turn) if they are all going to be assigned the same priority.

9 If redefining or deleting a priority for a type of production request **already** listed in the **PR Type Value-Priority** list, **single-click** on that PR type in the **Value-Priority** list.

- The PR type is highlighted.
- It is possible to highlight multiple PR types (by **single-clicking** on each one in turn while holding down either the **Shift** key or the **Ctrl** key) if the same action is going to be taken with respect to all of them.

- The **Find** button provides a means of performing a keyword search of the **PR Type Value-Priority** list.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **PR Type Value-Priority** list that has the search text is highlighted.)
  - Click on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

10 If defining or modifying (not deleting) a priority, click on the up/down arrow buttons to the right of the **Priority** field until the desired priority value is displayed in the **Priority** field.

- Alternatively, in the **Priority** field enter:

  <priority value>

- The acceptable range for the priority is from 1 to 10.

11 **Single-click** on the appropriate button from the following selections:

- **Add** - to approve a priority for an additional PR type and display the selected PR type and priority in the **PR Type Value-Priority** list at the left center of the GUI.
- **Modify** - to approve a revised priority for the selected PR type and display the PR type and modified priority in the **PR Type Value-Priority** list.
- **Delete** - to delete the priority for the selected PR type and remove the PR type and priority from the **PR Type Value-Priority** list.

12 Repeat Steps 7 through 11 as necessary until all PR Type priorities (as shown in the **PR Type Value-Priority** field) are correct.

13 If defining or modifying a priority for a type of user, first **single-click** on the **User Type** button.

- The different types of users are displayed in the **Type List** field at the bottom left of the GUI.
- Options are: **Operator**, **DAAC Manager**, **Scientist**, and **Researcher**.

14 If defining a priority for a type of user **not** currently listed in the **User Type Value-Priority** list, **single-click** on that user type in the **Type List** field.

- The user type is highlighted.
- It is possible to highlight multiple user types (by **single-clicking** on each one in turn).

15 If redefining or deleting a priority for a user type **already** listed in the **User Type Value-Priority** list, **single-click** on that user type in the **Value-Priority** list.

- The user type is highlighted.
- It is possible to highlight multiple user types (by **single-clicking** on each one in turn while holding down either the **Shift** key or the **Ctrl** key).

- The **Find** button provides a means of performing a keyword search of the **User Type Value-Priority** list.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **User Type Value-Priority** list that has the search text is highlighted.)
  - Click on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

16 If defining or modifying (not deleting) a priority, click on the up/down arrow buttons to the right of the **Priority** field until the desired priority value is displayed in the **Priority** field.

- Alternatively, in the **Priority** field enter:

  <priority value>
- The acceptable range for the priority is from 1 to 10.

17 **Single-click** on the appropriate button from the following selections:

- **Add** - to approve a priority for an additional user type and display the selected user type and priority in the **User Type Value-Priority** list near the center of the GUI.
- **Modify** - to approve a revised priority for the selected user type and display the user type and modified priority in the **User Type Value-Priority** list.
- **Delete** - to delete the priority for the selected user type and remove the user type and priority from the **User Type Value-Priority** list.

18 Repeat Steps 13 through 17 as necessary until all user type priorities (as shown in the **User Type Value-Priority** field) are correct.

19 If defining a priority for a type of PGE, first **single-click** on the **PGE Type** button.

- The different types of PGEs are displayed in the **Type List** field at the bottom left of the GUI.
- All PGEs currently registered in the PDPS database (for the mode in which the **Production Strategies** GUI was brought up) are listed.

20 If defining a priority for a type of PGE **not** currently listed in the **PGE Type Value-Priority** list, **single-click** on that PGE type in the **Type List** field.

- The PGE type is highlighted.
- It is possible to highlight multiple PGE types (by **single-clicking** on each one in turn).

21 If redefining or deleting a priority for a PGE type **already** listed in the **PGE Type Value-Priority** list, **single-click** on that PGE type in the **Value-Priority** list.

- The PGE type is highlighted.
- It is possible to highlight multiple PGE types (by **single-clicking** on each one in turn while holding down either the **Shift** key or the **Ctrl** key).

- The **Find** button provides a means of performing a keyword search of the **PGE Type Value-Priority** list.
  - A keyword search is performed by typing relevant text in the text entry box to the right of the **Find** button then clicking on the **Find** button. (The first entry in the **PGE Type Value-Priority** list that has the search text is highlighted.)
  - Click on the **Find** button again (as necessary) to highlight additional occurrences of the search text.

22 If defining or modifying (not deleting) a priority, click on the up/down arrow buttons to the right of the **Priority** field until the desired priority value is displayed in the **Priority** field.

- Alternatively, in the **Priority** field enter:

  <priority value>

- The acceptable range for the priority is from 1 to 10.

23 **Single-click** on the appropriate button from the following selections:

- **Add** - to approve a priority for an additional PGE type and display the selected PGE type and priority in the **PGE Type Value-Priority** list near the center of the GUI.
- **Modify** - to approve a revised priority for the selected PGE type and display the PGE type and modified priority in the **PGE Type Value-Priority** list.
- **Delete** - to delete the priority for the selected PGE type and remove the PGE type and priority from the **PGE Type Value-Priority** list.

24 Repeat Steps 19 through 23 as necessary until all PGE type priorities (as shown in the **PGE Type Value-Priority** field) are correct.

25 In the **Weight** field below the **PR Type** button enter:

<weight>

- The acceptable range for weights is from 1 to 100.
- The **Total Weight** field displays updated totals of all weighting factors as they are entered.
- When entering weights for the **PR Type**, **User Type**, **PGE Type**, and **Production Request Editor** factors, relative values can be entered without regard to whether the values in the four categories add up to 100. The **Normalize** button provides a means of eventually ensuring that the total of all four categories equals 100.
- The assigned weight in each category is multiplied by the priority for each type. To maintain a high priority (low number, such as one), assign a low weight; to ensure a low priority, assign a relatively high weight.

26 In the **Weight** field below the **User Type** button enter:

<weight>

27 In the **Weight** field below the **PGE Type** button enter:

<weight>

28 In the **Production Request Editor** field enter:

<weight>

- The priority to which the weight is applied is the priority assigned using the Production Request Editor when a production request is created.

29 **Single-click** on the **Normalize** button.

- The Planning Subsystem adjusts all weighting factors to produce a total weight of 100 (as displayed in the **Total Weight** field).

30 If it is necessary to change the priority of jobs that have been waiting in the Production Queue for more than a day, in the **Late Start Delta Priority** field enter:

<priority value>

- The range for the Late Start Delta Priority is from 1 to 100.

31 To start the process of saving the production strategy execute the following menu path:

File → Save As

- A **Save As** window similar to the **File Selection** window is displayed.

32 In the **Save As** field enter:

<strategy file name>

33 **Single-click** on the appropriate button from the following selections:

- **OK -** to accept the file name in the **Save As** field.
  - The **Save As** window is dismissed.
  - The production strategy is saved with the specified file name.
- **Cancel** - to dismiss the. **Save As** window without saving the production strategy.

NOTE: The **Help** button at the bottom of **Production Strategies Save As** window is non-functional (does not work).

34 To start the process of modifying or creating another production strategy execute the following menu path:

File → New

- Return to Step 2 to create another new production strategy.
- Return to Step 3 to modify an existing production strategy.

35 To start the process of exiting from the **Production Strategies** GUI execute the following menu path:

File → Exit

- A **Do you really want to exit?** dialogue box is displayed.

36 **Single-click** on the appropriate button from the following selections:

- **OK** - to exit from the **Production Strategies** GUI.
  - The **Production Strategies** GUI is dismissed.
- **Cancel** - to return to the **Production Strategies** GUI.

Table 13.3-3. Define or Modify a Production Strategy - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Launch **Production Strategies** GUI	Use procedure in Section 13.3.1
2	**File → New**	**single-click**
3	**<default priority>** (for **PR Type**) (if applicable)	**enter text**
4	**<default priority>** (for **User Type**) (if applicable)	**enter text**
5	**<default priority>** (for **PGE Type**) (if applicable)	**enter text**
6	**PR Type** button	**single-click**
7	**<PR type>** (from the **Type List** field)	**single-click**
8	**<priority value>** (**Priority** field)	**enter text** or **single-click** as necessary
9	**Add** button	**single-click**
10	Repeat Steps 7 through 9 for additional PR types if applicable	
11	**User Type** button	**single-click**
12	**<user type>** (**Type List** field)	**single-click**
13	**<priority value>** (**Priority** field)	**enter text** or **single-click** as necessary
14	**Add** button	**single-click**
15	Repeat Steps 12 through 14 for additional user types if applicable	
16	Select **PGE Type** button	**single-click**
17	Select **<PGE type>** from the **Type List** field	**single-click**
18	Enter **<priority value>** in the **Priority** field	**enter text** or **single-click** as necessary
19	Select **Add**	**single-click**
20	Repeat Steps 17 through 19 for additional PGE types if applicable	
21	**<weight>** (**PR Type**)	**enter text**
22	**<weight>** (**User Type**)	**enter text**
23	**<weight>** (**PGE Type**)	**enter text**
24	**<weight>** (**Production Request Editor**)	**enter text**

Table 13.3-3. Define or Modify a Production Strategy - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
25	**Normalize** button	**single-click**
26	**<priority value>** (for **Late Start Delta Priority**) (if applicable)	**enter text**
27	**File** → **Save As**	**single-click**
28	**<strategy file name>** (**Save As** field)	**enter text**
29	**Ok** button	**single-click**
30	**File** → **Exit** (when applicable)	**single-click**
31	**OK** button (when applicable)	**single-click**

13.3.3 Review the Current Active Strategy

The Production Planner uses the Production Strategies GUI to review the production strategy applied to the currently active production plan. Table 13.3-4 presents (in a condensed format) the steps required to review the current active strategy. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Launch the **Production Strategies** GUI (refer to Section 13.3.1).

- The **Production Strategies** GUI is displayed.

2 Execute the following menu path:

Options → **activeStrategy**

- The **Active Production Strategy** window is displayed.

3 Review the data displayed in the **Active Production Strategy** window:

- **PR Type.**
  - **Weight.**
  - **Default** [value].
  - [Individual PR types and values if any are listed.]
- **User Type.**
  - **Weight.**
  - **Default** [value].
  - [Individual user types and values if any are listed.]
- **PGE Type.**
  - **Weight.**
  - **Default** [value].
  - [Individual PGE types and values if any are listed.]

- **Production Request Editor Weight.**
- **Late Start Delta Priority.**

4 To exit from the **Active Production Strategy** window and return to the **Production Strategies** GUI **single-click** on the **Cancel** button.
- The **Active Production Strategy** window is dismissed.
- The **Production Strategies** GUI is displayed.

Table 13.3-4. Review the Current Active Strategy - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Strategies** GUI	Use procedure in Section 13.3.1
2	**Options → activeStrategy**	**single-click**
3	Observe the production strategy information	**read text**
4	**Cancel** button	**single-click**

13.3.4 Delete a Production Strategy

The Production Planner uses the Production Strategies GUI to delete production strategies that are no longer needed. Table 13.3-5 presents (in a condensed format) the steps required to delete a production strategy. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Launch the **Production Strategies** GUI (refer to Section 13.3.1).
- The **Production Strategies** GUI is displayed.

2 **Single-click** the option button associated with the **Production Strategies** field, then highlight (in the option menu) the name of the production strategy to be deleted.
- Data pertaining to the selected production strategy are displayed in the applicable fields of the **Production Strategies** GUI.
- Alternatively, it is possible to execute the following menu path: **File → Open**, select the desired production strategy from the list on the **Open** window, and **single-click** on the **Ok** button to open the production strategy.

3 Execute the following menu path:

Edit → Delete

- A production strategy deletion confirmation dialogue box is displayed requesting confirmation of the decision to delete the production strategy.

4 Click on the appropriate button from the following selections:

- **OK -** to confirm deletion of the production strategy and dismiss the dialogue box.
  - The deletion confirmation dialogue box is dismissed.
  - The production strategy is deleted.
- **Cancel** - to preserve the production strategy and dismiss the dialogue box.
  - The deletion confirmation dialogue box is dismissed.
  - The production strategy is not deleted.

5 To start the process of exiting from the **Production Strategies** GUI execute the following menu path:

File → Exit

- A **Do you really want to exit?** dialogue box is displayed.

6 **Single-click** on the appropriate button from the following selections:

- **OK** - to exit from the **Production Strategies** GUI.
  - The **Production Strategies** GUI is dismissed.
- **Cancel** - to return to the **Production Strategies** GUI.

Table 13.3-5. Delete a Production Strategy - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Production Strategies** GUI	Use procedure in Section 13.3.1
2	**<stra0tegy name>** (from **Production Strategies** option list)	**single-click**
3	**Edit → Delete**	**single-click**
4	**OK** button	**single-click**
3	**File → Exit** (when applicable)	**single-click**
5	**OK** button	**single-click**

13.4 Creating/Deleting Production Plans

The planning process involves the Production Planner preparing monthly and weekly production plans as well as a daily production schedule from the most current weekly plan. Although production planning varies from DAAC to DAAC, the following guidelines are generally applicable:

- Monthly plans.
  - Developed for the coming month and one or two months in advance.
  - Produced, reviewed, updated, published and distributed approximately two weeks before the beginning of the month.
  - Plan for the coming month is used to establish a baseline against which production targets can be measured.

- Weekly plans.
  - Produced, reviewed, updated, published and distributed approximately five days before the beginning of the coming week.
  - Used to produce a baseline for comparison of planned vs. actual production results.
- Daily plan or schedule.
  - Produced each day for the next processing day.
  - Developed from the current weekly plan, adjusted to reflect the actual processing accomplished and the actual resources available at the time the daily schedule is generated.
  - The daily plan is the type of plan that is implemented through the **Planning Workbench** GUI.
  - If there is a very large processing volume, it may be advisable to divide the daily plan into multiple plans; e.g., one for each shift or one for each four-hour period.

During normal processing, when reasonably accurate predictions of the processing time for the PGEs are available, the processing schedule should result in a reasonably accurate prediction of when data products will be generated. However, during abnormal situations (e.g., equipment failure), what is actually accomplished could depart significantly from the plan. In such situations, the Production Planner may choose to develop new plans to reflect current events. This process is known as “replanning.”

Table 13.4-1 provides an Activity Checklist for activities related to the creation of Production Plans.

Table 13.4-1. Production Plans - Activity Checklist

Order	Role	Task	Section	Complete?
1	Production Planner	Launch the Planning Workbench and Planning Timeline GUIs	(P) 13.4.1	
2	Production Planner	Create a New Production Plan	(P) 13.4.2	
3	Production Planner	Delete a Production Plan	(P) 13.4.3	
4	Production Planner	Review a Plan Timeline	(P) 13.4.4	

13.4.1 Launch the Planning Workbench and Planning Timeline GUIs

The Planning Workbench and Planning Timeline GUIs are invoked from a UNIX command line prompt. Table 13.4-2 presents (in a condensed format) the steps required to launch the Planning Workbench and Planning Timeline GUIs. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Access a terminal window logged in to the Planning/Management Workstation host.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 In the terminal window, at the command line, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1 - Science Software Integration and Test (SSI&T)
  - TS2 - New Version Checkout
  - OPS - Normal Operations
- "utilities" is the directory containing the Planning Subsystem start-up scripts.

3 Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

NOTE: The selection of a start-up script for launching the **Planning Workbench** GUI depends on the number of DPRs in the PDPS database. If the number is very high (~4000), the **EcPlSomeStart** script is used to start the underlying processes (Message Handler, System Name Server, and Resource Model). Then additional scripts are used to start the **Planning Workbench** GUI and the **Production Planning Timeline** GUI. If the number of DPRs in the PDPS database is not very high (less than about 4000), the **EcPlAllStart** script is used to start the underlying processes, **Planning Workbench** GUI and **Production Planning Timeline** GUI.

4 If the number of DPRs in the PDPS database is not large (less than about 4000), start the Planning Workbench-related applications by entering:

EcPlAllStart <MODE> <Application_id>

- The System Name Server and Resource Model are launched.
- The **Message Handler** GUI is displayed.
- Eventually, the **Planning Workbench** GUI is displayed.
- Then the **Production Planning Timeline** GUI is displayed.
  - The **Production Planning Timeline** GUI usually occupies the entire screen when it is initially displayed.

- The **application_id** or **MSGSRV_ID** is a number from 1 to 5. It identifies the message service in use so messages can be directed to the proper message handler GUI. Consequently, it is a good idea to use the same application_id consistently during a production planning session.
- Go to Step 8.

5 If the number of DPRs in the database is large (~4000 or more), start the Message Handler, System Name Server, and Resource Model by entering:

EcPlSomeStart <MODE> <Application_id>

- The **Message Handler** GUI is displayed.
- Wait until the Resource Model is completely up before proceeding with starting the **Planning Workbench** GUI.

6 If the number of DPRs in the database is large (~4000 or more), start the **Planning Workbench** GUI by entering:

EcPlWbStart <MODE> <Application_id>

- The **Planning Workbench** GUI is displayed.

7 If the number of DPRs in the database is large (~4000 or more), start the **Production Planning Timeline** GUI by entering

EcPlTlStart <MODE> <Application_id>

- The **Production Planning Timeline** GUI is displayed.

8 If the **Production Planning Timeline** GUI is occupying the whole screen, either **single-click** on the “minimize” icon in the upper right corner of the GUI or adjust the window size and the view of the timeline as necessary using the mouse.

- Grab a corner of the timeline window with the cursor and resize the window as desired.

Table 13.4-2. Launch the Planning Workbench and Planning Timeline GUIs - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	Set environment variables if necessary	**enter text, press Enter**
4	**EcPlAllStart <MODE>** <**application_id**>	**enter text, press Enter**

13.4.2 Create a New Production Plan

The Production Planner uses the Planning Workbench when creating a plan for production data processing at the DAAC. The Planning Workbench GUI provides the means by which the Production Planner selects specific PRs whose DPRs are to be run. The planning tool provides a forecast of the start and completion times of the jobs based upon historical experience in running these PGEs (as determined during the SSI&T process). Through the planning tool, when the generated plan is "activated," the information included in the plan is transferred to the Data Processing Subsystem and loaded into the Platinum AutoSys tool where production processing is managed.

Table 13.4-3 presents (in a condensed format) the steps required to create a new Production Plan. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Launch the Planning Workbench (refer to Section 13.4.1).

- The **Planning Workbench** and **Planning Timeline** GUIs are displayed.

2 Execute the following menu path on the **Planning Workbench GUI**:

File → **New**

- The **New Plan** window is displayed.

3 In the **Plan Names** field enter:

<plan name>

- Name is displayed in **Plan Names** field.
- Status is **Candidate**.

4 **Single-click** on the appropriate button from the following selections:

- **Ok** - to accept the file name in the **Plan Names** field.
  - The **New Plan** window is dismissed.
  - The production plan is saved with the specified file name.
- **Apply** - to save the production plan without dismissing the **New Plan** window.
  - The production plan is saved with the specified file name.
- **Cancel** - to dismiss the **New Plan** window without saving the production plan.

NOTE: The **Help** buttons at the bottom of various **Planning Workbench** windows are non-functional (do not work).

5 If applicable, **single-click** on the option button associated with the **Strategy** field, then select the desired production strategy from the option menu.

6 If applicable, in the **Comment** field enter:

<comment>

7 Select a Production Request to be scheduled by **single-clickin**g on the request line and **single-clicking** on **schedule/unschedule** up and down arrows to move the Production Request from the **Unscheduled** panel to the **Scheduled** panel or vice versa.

- Production Request will appear in the appropriate panel.
- Operator can **single-click** on multiple Production Requests to add them all at the same time.
- If the PR depends upon another PR that is not scheduled, then the following message is displayed: "Production Request '*xxx*' must be selected before DPR '*yyy*' of Production Request '*zzz*' can be scheduled."
- If all DPRs associated with a PR have been run, the PR cannot be rescheduled.
- If processing of the currently active plan is to be continued when the new plan is activated, include the PR(s) for the currently active plan in the new plan.
- In the **Scheduled** list, items with the prefix "GE_" are resource reservations (also called "ground events").
- All ground events are automatically scheduled with any plan; therefore, the ground events are normally displayed in the Scheduled list.
- Whenever a plan is activated, the ground events are activated as well as the DPRs associated with the specified PRs.
- If a ground event appears in the **Unscheduled** list, the ground event has lost allocations.

8 If the priority of any PR in the **Scheduled** list needs to be changed, perform Steps 9 through 13; otherwise go to Step 14.

9 **Single-click** on the PR entry in the **Scheduled** list to highlight it.

10 **Single-click** on the **Prioritize** button.

- The **Priority popup** window is displayed.

11 In the **Production Request(s) priority** field enter:

<priority value>

12 **Single-click** on the appropriate button from the following selections:

- **OK** - to accept the new priority in the **Production Request(s) priority** field.
  - The **Priority popup** window is dismissed.
  - The new priority for the PR is saved in the database.
- **Cancel** - to dismiss the **Priority popup** window without saving the new priority.

13 Repeat Steps 9 through 12 for any additional PR(s) needing a change of priority.

14 Execute the following menu path:

File → Save As

- The **Save Plan** window is displayed.

15 If the **plan name** for the production plan is not displayed in the **Plan Names** field, enter:

<plan name>

16 **Single-click** on the appropriate button from the following selections:

- **Ok** - to accept the file name in the **Plan Names** field.
  - The **Save Plan** window is dismissed.
  - The production plan is saved with the specified file name.
  - The **Planning Workbench** GUI is displayed.
  - The **Plan Name** is displayed.
  - The **Status** displayed is **Candidate**.
- **Apply** - to save the production plan without dismissing the **Save Plan** window.
  - The production plan is saved with the specified file name.
- **Cancel** - to dismiss the **Save Plan** window without saving the production plan.

17 If the plan is to be activated immediately, perform Steps 18 and 19; otherwise go to Step 20.

18 **Single-click** on the **Activate** button.

- A **Confirm Activation** dialogue box is displayed.

19 **Single-click** on the appropriate button from the following selections:

- **Yes** - to activate the plan.
  - The **Confirm Activation** dialogue box is dismissed.
  - The new plan is activated.
  - The time of plan activation is displayed next to **Rollover Time** on the **Planning Workbench** GUI.
  - When a plan is activated the Data Processing Requests (DPRs) associated with the planned PRs are transferred to Job Management in the Data Processing Subsystem.
  - Once its data dependencies have been satisfied, each DPR is “released” to be run as processing resources become available.
  - Activating a new plan causes the current active plan to get "replanned over" by the selected plan.
- **No –** to dismiss the **Confirm Activation** dialogue box without activating the plan.

20 If it is desired to baseline the plan, perform Steps 21 through 25; otherwise go to Step 26.

21 **Single-click** on the **Baseline** button.
- A confirmation dialogue box containing the message **The current plan is <Plan Name>. Do you wish to baseline it?** is displayed.
- Clicking on the **Baseline** button records the plan and the time of baselining.
- A baseline plan can be used as a point of comparison with which to compare future plans and results.

22 If it is desired to baseline the plan, **single-click** on the appropriate button from the following selections:
- **Yes** - to baseline the plan.
  - The confirmation dialogue box containing the message **The current plan is <Plan Name>. Do you wish to baseline it?** is dismissed.
  - The plan is baselined.
- **No** – to dismiss the confirmation dialogue box containing the message **The current plan is <Plan Name>. Do you wish to baseline it?** without baselining the plan.

23 Execute the following menu path:

File → Save As

- The **Save Plan** window is displayed.

24 If the **plan name** for the production plan is not displayed in the **Plan Names** field, enter:

<plan name>

25 **Single-click** on the appropriate button from the following selections:
- **Ok** - to accept the file name in the **Plan Names** field.
  - The **Save Plan** window is dismissed.
  - The production plan is saved with the specified file name.
- **Apply** - to save the production plan without dismissing the **Save Plan** window.
  - The production plan is saved with the specified file name.
- **Cancel** - to dismiss the **Save Plan** window without saving the production plan as a baseline plan.

26 Repeat Steps 1 through 25 to perform additional production planning activities.

27 To view the **Planning Timeline** perform the procedure in Section 13.4.3.

28 To quit the **Planning Workbench** GUI when production planning is complete execute the following menu path:

File → Exit

29 After quitting the **Planning Workbench** GUI **single-click** in the UNIX window used to start the **Planning Workbench** GUI.

- The Message Handler, System Name Server, and Resource Model should be shut down to eliminate unneeded processes and allow other operators to gain access to the Planning Workbench if necessary.

30 Shut down Planning Workbench-related applications by entering:

EcPlSlayAll <MODE> <Application_id>

- The following Planning Workbench-related applications shut down:
  – Planning Workbench (if it has not already been shut down).
  – Planning Timeline (if it has not already been shut down).
  – Message Handler.
  – System Name Server.
  – Resource Model.

31 Obtain a list of active processes in the specified mode by entering:

ps -ef | grep <MODE>

- A list of active processes in the specified mode is displayed.
- If an error message is received when **ps -ef | grep <MODE>** is entered, enter:

  ps -auxwww | grep <MODE>

32 Examine the list of processes running in the specified mode to determine whether the Message Handler, System Name Server, and Resource Model processes have actually been shut down.

- None of the following processes should be active:
  – EcPlWb
  – EcPlTl
  – EcPlMsh
  – EcPlSns
  – EcPlRm

33 If any of the specified processes [especially the Message Handler, System Name Server, and/or Resource Model process(es)] is/are still active, terminate the active process(es) by entering:

kill -15 <PROCESS ID1> <PROCESS ID2> <…> <PROCESS IDx>

34 Repeat Steps 31 through 33 as necessary.

Table 13.4-3. Create a New Production Plan - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Planning Workbench**	Use procedure in Section 13.4.1
2	**File → New**	**single-click**
3	**<plan name>**	**enter text**
4	**Ok** button	**single-click**
5	**<strategy>** (from **Strategy** button)	**single-click**
6	**<Production Request(s)>** to schedule/unschedule	**single-click**
7	Either **Schedule** button or **Unschedule** button as applicable	**single-click**
8	**File → Save As**	**single-click**
9	**<plan name>**	**enter text**
10	**Ok** button	**single-click**
11	**Activate** button (if applicable)	**single-click**
12	**Yes** button (if applicable)	**single-click**
13	**Baseline** button (if applicable)	**single-click**
14	**Yes** button (if applicable)	**single-click**
15	**File → Save As** (if applicable)	**single-click**
16	**<plan name>** (if applicable)	**enter text**
17	**Ok** button (if applicable)	**single-click**
18	View the **Planning Timeline** if desired	Use procedure in Section 13.4.3
19	**File → Exit** (when applicable)	**single-click**
20	**EcPlSlayAll <MODE> <Application_id>**	**enter text, press Enter**
21	**ps -ef | grep <MODE>**	**enter text, press Enter**
22	**kill -15 <PROCESS ID1> <PROCESS ID2> <…> <PROCESS IDx>** to terminate active process(es) (if necessary)	**enter text, press Enter**

13.4.3 Delete a Production Plan

The Production Planner uses the Planning Workbench GUI to delete production plans that are no longer needed. Table 13.4-4 presents (in a condensed format) the steps required to delete a production plan. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Launch the **Planning Workbench** GUI (refer to Section 13.4.1).

- The **Planning Workbench** GUI is displayed.

2 Execute the following menu path:

File → Open

- The **Open Plan** window is displayed.
- A list of Production Plans is displayed in the **Open Plan** window.

3 Select (highlight) the production plan to be deleted by **single-clicking** on the corresponding name in the list of plans.

- The plan name is displayed in the **Plan Names** field of the **Open Plan** window.

4 **Single-click** on the **OK** button.

- The **Open Plan** window is dismissed.
- The plan information is displayed on the **Planning Workbench** GUI.

5 Execute the following menu path:

File → Delete

- The production plan is deleted.

6 To quit the **Planning Workbench** GUI execute the following menu path:

File → Exit

7 After quitting the **Planning Workbench** GUI **single-click** in the UNIX window used to start the **Planning Workbench** GUI.

- The Message Handler, System Name Server, and Resource Model should be shut down to eliminate unneeded processes and allow other operators to gain access to the Planning Workbench if necessary.

8 Shut down Planning Workbench-related applications by entering:

EcPlSlayAll <MODE> <Application_id>

- The following Planning Workbench-related applications shut down:
  - Planning Workbench (if it has not already been shut down).
  - Planning Timeline (if it has not already been shut down).
  - Message Handler.
  - System Name Server.
  - Resource Model.

9 Obtain a list of active processes in the specified mode by entering:

ps -ef | grep <MODE>

- A list of active processes in the specified mode is displayed.

- If an error message is received when **ps -ef | grep <MODE>** is entered, enter:

  ps -auxwww | grep <MODE>

10 Examine the list of processes running in the specified mode to determine whether the Message Handler, System Name Server, and Resource Model processes have actually been shut down.

- None of the following processes should be active:
  - EcPlWb
  - EcPlTl
  - EcPlMsh
  - EcPlSns
  - EcPlRm

11 If any of the specified processes [especially the Message Handler, System Name Server, and/or Resource Model process(es)] is/are still active, terminate the active process(es) by entering:

kill -15 <PROCESS ID1> <PROCESS ID2> <…> <PROCESS IDx>

12 Repeat Steps 9 through 11 as necessary.

Table 13.4-4. Delete a Production Plan - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Launch **Planning Workbench** GUI	Use procedure in Section 13.4.1
2	**File → Open**	**single-click**
3	**<plan name>** (from **Production Requests** list)	**single-click**
4	**OK** button	**single-click**
5	**File→Delete**	**single-click**
6	**File → Exit**	**single-click**
7	**EcPlSlayAll <MODE> <Application_id>**	**enter text, press Enter**
8	**ps -ef | grep <MODE>**	**enter text, press Enter**
9	**kill -15 <PROCESS ID1> <PROCESS ID2> <…> <PROCESS IDx>** to terminate active process(es) (if necessary)	**enter text, press Enter**

13.4.4 Review a Plan Timeline

Table 13.4-5 presents (in a condensed format) the steps required to review planning timelines. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Launch the **Planning Workbench** and **Timeline** GUIs (refer to Section 13.4.1).

- The **Planning Timeline** GUI is displayed.
- If you have previously saved a configuration file, you may load it by executing the following menu path:

  File → Load Configuration

  – Change directory in GUI box, if necessary.
- Otherwise continue with Step 2.

2 Execute the following menu path:

File → Open Plan

- The **Open Plan** window is displayed with a list of plans.

3 **Single-click** on the name (in the **Items** list) of the plan to be reviewed.

4 **Single-click** on the appropriate button from the following selections:

- **OK** - to open the selected plan and dismiss the **Open Plan** window.
  – The timeline for the specific plan is displayed.
  – Name is displayed in the Title bar.
- **Apply** - to open the selected plan without dismissing the **Open Plan** window.
  – The timeline for the specific plan is displayed.
  – Name is displayed in the Title bar.
- **Cancel** - to dismiss the **Open Plan** window without opening any plan.

NOTE: The **Help** buttons at the bottom of various **Production Planning Master Timeline** windows are non-functional (do not work).

5 Observe the production plan information displayed on the timeline GUI.

6 If a different time scale (start and end dates and times) is desired, perform Steps 7 through 10; otherwise, go to Step 11.

7 Execute the following menu path:

Time → Change Plan Window

- The **plan window edit** window appears with default times.

8 In the **Plan Win Start** fields enter the start date and time in the following formats:

<DD MMM YYYY> <hh:mm:ss>

9 In the **Plan Win End** fields enter the end date and time in the following formats:

<DD MMM YYYY> <hh:mm:ss>

10 **Single-click** on the appropriate button from the following selections:

- **OK** - to accept the changes and dismiss the **plan window edit** window.
- **Apply** - to accept the changes without dismissing the **plan window edit** window.
- **Cancel** - to cancel the changes and dismiss the **plan window edit** window.

11 If a different time span is desired, **single-click** on the **Show** pushbutton and select one of 11 time increments between **5 min** and **168 hrs** for the timeline scale.

- The entry "**other**" has no purpose at this time.

12 If no resources are displayed on the GUI or if different resources should be displayed, perform Steps 13 through 19; otherwise, go to Step 20.

13 Execute the following menu path:

Display → Change resources

- **Resource edit** window with lists of **Available Resources** and **Viewed Resources** is displayed.

14 **Single-click** resource(s) in desired list.

- Multiple resources may be selected by single-clicking each desired resource.

15 **Single-click** either **Add** or **Del**.

- **Add** to move the resource(s) from the **Available** list to the **Viewed** list.
- **Del** to remove items from the **Viewed** List.

16 To change the order in which resources are displayed on the timeline, **single-click** on an item in the **Viewed Resources** list.

17 To change the order in which resources are displayed on the timeline, **single-click** on the **up** or **down** arrow(s) as appropriate.

- Selected resource moves up or down in order on the list.

18 Repeat Steps 16 and 17 as necessary.

19 **Single-click** on the appropriate button from the following selections:

- **OK** - to accept the changes and dismiss the **Resource edit** window.
- **Apply** - to accept the changes without dismissing the **Resource edit** window.
- **Cancel** - to cancel the changes and dismiss the **Resource edit** window.

20 If different color coding of the timeline is desired, perform Steps 21 through 25; otherwise, go to Step 26.

21 Execute the following menu path:

Display → Change colors

- Color grid appears with a list of Production Requests.

22 **Single-click** on the Production Request name.

23 **Single-click** on the desired color for the Production Request.

- New color appears on the horizontal bar between color and Production Request selections.

24 Repeat Steps 22 and 23 as necessary.

25 **Single-click** on the appropriate button from the following selections:

- **OK** - to accept the changes and dismiss the **Color Selections** window.
- **Apply** - to accept the changes without dismissing the **Color Selections** window.
- **Cancel** - to cancel the changes and dismiss the **Color Selections** window.

26 Observe the production plan information displayed on the timeline GUI.

27 If desired, save the current configuration as a file by executing the following menu path:

File → Save Configuration

28 **Single-click** on the appropriate button from the following selections:

- **OK** - to save the configuration and dismiss the **Color Selections** window.
- **Apply** - to save the configuration without dismissing the **Color Selections** window.
- **Cancel** - to dismiss the **Color Selections** window without saving the configuration.

29 To exit the **Planning Master Timeline** GUI, execute the following menu path:

File → Exit

- The **Planning Master Timeline** GUI is dismissed.

Table 13.4-5. Review a Plan Timeline - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Launch **Planning Workbench** and **Timeline**	Use procedure in Section 13.4.1
2	**File → Load Configuration**) (optional)	**single-click**
3	**File → Open Plan**	**single-click**

Table 13.4-5. Review a Plan Timeline - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
4	**<plan name>**	**single-click**
5	**Ok** button	**single-click**
6	Observe the production plan information	**read text**
7	**Time → Change Plan Window**	**single-click**
8	**<DD MMM YYYY> <hh:mm:ss>** (in **Plan Win Start** fields)	**enter text**
9	**<DD MMM YYYY> <hh:mm:ss>** (in **Plan Win End** fields)	**enter text**
10	**Ok** button	**single-click**
11	**<time span>** (from **Show** pushbutton)	**single-click**
12	**Display → Change resources** (if applicable)	**single-click**
13	**<resources>** (if applicable)	**single-click**
14	**Add** button (if applicable)	**single-click**
15	**<viewed resource>** to be reordered (if applicable)	**single-click**
16	**<up>** arrow or **<down>** arrow (as necessary to reorder viewed resources) (if applicable)	**single-click**
17	**Ok** button (if applicable)	**single-click**
18	**Display → Change colors** (if applicable)	**single-click**
19	**<Production Request>** (if applicable)	**single-click**
20	**<color>** (for Production Request) (if applicable)	**single-click**
21	**Ok** button (if applicable)	**single-click**
22	Observe the production plan information	**read text**
23	**File → Save Configuration** (if applicable)	**single-click**
24	**<file name>** (if applicable)	**enter text**
25	**Ok** button (if applicable)	**single-click**
26	**File → Exit** (when applicable)	**single-click**

13.5 Cleaning the PDPS Database and Science Processing Disks

At DAACs with a heavy data processing load it is essential to perform PDPS database cleanup and virtual-computer “garbage collection” on a regular basis. The frequency of cleanup is determined by each DAAC’s needs. Failure to perform cleanup is likely to lead to performance problems with the Planning and Data Processing Subsystems.

It is recommended that the database cleanup and “garbage collection” processes be run in the following order:

- Database cleanup.
- “Garbage collection” on the virtual computer(s).

- Database cleanup (again).

If not already done, it is possible to create a script to run the preceding processes. If it is necessary to run particular scripts individually, refer to the sections that follow.

NOTE: To the extent possible the scripts mentioned in this section should be run when the system is relatively quiet; i.e., when no PRs/DPRs are being created and no jobs are running in AutoSys.

Table 13.5-1 provides an Activity Checklist for activities related to cleaning the PDPS database and science processing disks.

Table 13.5-1. Cleaning the PDPS Database and Science Processing Disks - Activity Checklist

Order	Role	Task	Section	Complete?
1	Production Planner Database Administrator	Clean the PDPS Database	(P) 13.5.1	
2	Production Planner Production Monitor	Perform Garbage Collection	(P) 13.5.2	
3	Production Planner Production Monitor	Run the Deletion Server Client	(P) 13.5.3	
4	Production Planner Production Monitor	Resolve PDPS Database and Science Processing Disk Content Discrepancies	(P) 13.5.4	
5	Production Planner Database Administrator	Save and/or Reset the PDPS Database	(P) 13.5.5	

13.5.1 Clean the PDPS Database

In the /usr/ecs/<MODE>/CUSTOM/utilities directory on the Planning/Management Workstation there is a script that can be run to clean up some tables in a PDPS database. When it runs, the script tries to delete applicable records in the following order:

- Data Processing Requests based on timeStamp completionState(SUCC_DEL).
- Production Requests that have no associated DPRs.
- Dynamic data granules that are not used by any DPR or by the Data Processing Subsystem.
- List of data granules that are not deleted because of Data Processing Subsystem usage.
- PGEs marked with a deleteFlag.
- Science Software that has no associated PGE.

The database cleaning script is invoked from a UNIX command line prompt. Table 13.5-2 presents (in a condensed format) the steps required to clean the PDPS database. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Access a terminal window logged in to the Planning/Management Workstation host.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

3 In the terminal window, at the command line prompt, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1 - Science Software Integration and Test (SSI&T)
  - TS2 - New Version Checkout
  - OPS - Normal Operations
- "utilities" is the directory containing the Planning Subsystem utility/start-up scripts.

4 Start the database cleaning script by entering:

EcPlDbClean <MODE> <database user> <database password> <database server> <months> <days>

- **<database user>** is the user name for logging in to interactive structured query language (isql).
- **<database password>** is the password for isql login.
- **<database server>** refers to the name of the PDPS database server (e.g., x0sps02_srvr).
- **<months>** is a number specifying the removal of records that are older than that number of months.
- **<days>** is an optional argument. It is a number that specifies the removal of records that are older than that number of days.
- Both **<months>** and **<days>** are taken into account by the cleaning script.
- When the process has finished, the following type of message is displayed:

  Log is written to /usr/ecs/OPS/CUSTOM/logs/EcPlDbClean.log

5 When the EcPlDbClean process has finished, enter:

cd ../logs

- Change directory to the directory containing the production planning log files (including EcPlDbClean.log).

6 At the command line prompt enter:

pg <filename>

- **<filename>** refers to the log file to be reviewed (e.g., EcPlDbClean.log).
- The first page of the log file is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **view, vi**) can be used to review the log file.

7 Review the log file to determine the results of running the script.

- To exit from **pg** at the **:** prompt enter:

  q

  – The command line prompt is displayed.

Table 13.5-2. Clean the PDPS Database - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**Single-click** or use procedure in Section 13.2.1
2	Set environment variables if necessary	**enter text, press Enter**
3	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
4	**EcPlDbClean** <**MODE**> <**database server**> <**database user**> <**database password**> <**months**> <**days**>	**enter text, press Enter**
5	**cd ../logs** (when the EcPlDbClean process has finished)	**enter text, press Enter**
6	**pg <filename>**	**enter text, press Enter**
7	Review the log file	**read text**

13.5.2 Perform Garbage Collection

In the /usr/ecs/*MODE*/CUSTOM/utilities directory on the Queuing Server there is a script (i.e., EcDpPrGarbageCollectorStart) that can be run to delete unneeded files from the science processing disks and update the PDPS database accordingly.

The actual executable invoked by the EcDpPrGarbageCollectorStart script is EcDpPrDeletionClient. However, the EcDpPrGarbageCollectorStart script differs from the EcDpPrDeletionClientStart script in the following three ways:

- Does not open a separate xterm window.
- Requires specification of a MACHINE_TO_COLLECT variable.
- Includes retry logic in case of database deadlock.

The garbage-collection script is invoked from a UNIX command line prompt. Table 13.5-3 presents (in a condensed format) the steps required to perform garbage collection. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Access a terminal window logged in to the Queuing Server host.

- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03**.
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

3 In the terminal window, at the command line prompt, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1 - Science Software Integration and Test (SSI&T)
  - TS2 - New Version Checkout
  - OPS - Normal Operations
- “utilities” is the directory containing the production processing utility/start-up scripts (e.g., EcDpPrAutosysStart).
- Note that the separate subdirectories under /usr/ecs apply to (describe) different operating modes.

4 Start the garbage collection script by entering:

EcDpPrGarbageCollectorStart <MODE> <machine> <retries> <interval>

- **<machine>** refers to the name of the virtual computer (e.g., x0spg11_vc) for which garbage collection is being requested.
  - The value entered is assigned to a MACHINE_TO_COLLECT variable.
- **<retries>** is the number of retries in case of database deadlock.
  - The default number of retries is five (5) [times].

- **<interval>** is the amount of time (in minutes) between retries.
  - The default interval is 30 [minutes].

5 Observe the results as the script runs.
- A UNIX command line prompt is displayed when the script has finished.

Table 13.5-3. Perform Garbage Collection - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Queuing Server host)	**Single-click** or use procedure in Section 13.2.1
2	Set environment variables if necessary	**enter text, press Enter**
3	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
4	**EcDpPrGarbageCollectorStart <MODE> <machine> <retries> <interval>**	**enter text, press Enter**
5	Observe script results	**read text**

13.5.3 Run the Deletion Server Client

In the /usr/ecs/*MODE*/CUSTOM/utilities directory on the Queuing Server there is a script (i.e., EcDpPrDeletionClientStart) that can be run to delete unneeded files from the science processing disks and update the PDPS database accordingly. The executable invoked by the script is EcDpPrDeletionClient.

The script for running the Deletion Server Client is invoked from a UNIX command line prompt. Table 13.5-4 presents (in a condensed format) the steps required to run the Deletion Server Client. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

1 Access a terminal window logged in to the Queuing Server host.
- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03**.
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

3 In the terminal window, at the command line prompt, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1 - Science Software Integration and Test (SSI&T)
  - TS2 - New Version Checkout
  - OPS - Normal Operations
- "utilities" is the directory containing the production processing utility/start-up scripts (e.g., EcDpPrAutosysStart).
- Note that the separate subdirectories under /usr/ecs apply to (describe) different operating modes.

4 Start the Deletion Server Client by entering:

EcDpPrDeletionClientStart <MODE>

5 Observe the results as the script runs.

- A UNIX command line prompt is displayed when the script has finished.

Table 13.5-4. Run the Deletion Server Client - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Queuing Server host)	**Single-click** or use procedure in Section 13.2.1
2	Set environment variables if necessary	**enter text, press Enter**
3	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
4	**EcDpPrDeletionClientStart <MODE>**	**enter text, press Enter**
5	Observe script results	**read text**

13.5.4 Resolve PDPS Database and Science Processing Disk Content Discrepancies

In the /usr/ecs/<MODE>/CUSTOM/utilities directory on the Queuing Server there is a script (i.e., EcDpPrRmFilesWOGranules.pl) that can be run to ensure consistency between the file references in various tables in the PDPS database and the files actually staged on the science processing disks. When it runs, the script performs the following functions:

- Generates a list of files with consistent references among various tables in the PDPS database for the specified mode.
- Checks for files on the disk that are not included in the list of files referenced in the PDPS database and either lists the inconsistent files or generates a script to delete them (as specified by the person running the script).

- Checks to determine whether the disk partitions referenced in the PDPS database actually exist on the disk(s).
  - If the disk partitions are not on the disk(s), the script removes references to them from the database.
- Removes all file references in the PDPS database that are not included in the list of files.
  - The script resets the PlRscDiskPartition according to the remaining DpPrDiskPartition entries (if specified by the person running the script).

The script for resolving PDPS database and science processing disk content discrepancies is invoked from a UNIX command line prompt. Table 13.5-5 presents (in a condensed format) the steps required to resolve PDPS database and science processing disk content discrepancies. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures.

NOTE: The **EcDpPrRmFilesWOGranules.pl** script should be run when the system is relatively quiet; i.e., when no jobs are running in AutoSys.

1 Access a terminal window logged in to the Planning/Management Workstation host.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 In the terminal window, at the command line, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1 - Science Software Integration and Test (SSI&T)
  - TS2 - New Version Checkout
  - OPS - Normal Operations
- “utilities” is the directory containing the Planning Subsystem utility/start-up scripts (e.g., EcDpPrAutosysStart).

3 Start the database cleaning script by entering:

EcDpPrRmFilesWOGranules.pl <database user> <database password> <MODE> <database server> fix [or **nofix**]

- **<database user>** is the user name for logging in to isql.
- **<database password>** is the password for isql login.
- **<database server>** refers to the name of the PDPS database server (e.g., x0sps02_srvr).

- **fix** (or **nofix**) specifies whether or not inconsistent files are removed from the science processing disks.
- An option is to add **>& fix.log** or **>& /usr/ecs/<MODE>/CUSTOM/logs/fix.log** to the end of the command line to direct the output of the script to a log file.
- For example:

  EcDpPrRmFilesWOGranules.pl pdps_role password1 OPS x0sps02_srvr nofix

4 Observe the results as the script runs.

- A UNIX command line prompt is displayed when the script has finished.

Table 13.5-5. Resolve PDPS Database and Science Processing Disk Content Discrepancies - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	**EcDpPrRmFilesWOGranules.pl <database user> <database password> <MODE> <database server> fix** [or **nofix**]	**enter text, press Enter**
4	Observe script results	**read text**

13.5.5 Save and/or Reset the PDPS Database

The utilities directory (/usr/ecs/<MODE>/CUSTOM/utilities) contains the following three scripts that are used in resetting the database:

- EcPlDbReset.
- EcPlDbList.
- EcPlDbSave.

Saving the database using the EcPlDbSave script produces one ASCII file (with a ".dat" extension) for each database table and writes each file to a specified directory. The saved data can be restored to the database by running the EcPlDbReset script with the name of the saved database as an argument.

Resetting the database involves clearing ("wiping out") the data in the database tables and loading values from a specified "saved database" file. It is important to take into consideration the consequences of resetting the database before performing the procedure. Resetting the database removes and replaces **all** Resource Definitions, Resource Reservations (Ground Events), Production Requests, Data Processing Requests, and Production Plans. There should be coordination with all affected parties, including the Resource Planner, Production Planner, and

Production Monitors concerning the effects of resetting the database as well as its after-effects (e.g., recreating resource definitions, resource reservations, and production requests).

Whenever the PDPS database is reset (not including database cleanup by running the EcPlDbClean script) it is also necessary to remove all PLS subscriptions in the Communications Subsystem (CSS) Subscription Server database (where the subscriber is Subscription Manager). Production personnel can remove the subscriptions using the Subscription Server GUI (EcSbSubServerGUI) if they have access to the GUI. Otherwise, they can request User Services personnel to remove the subscriptions.

As a result of removing the subscriptions, no subscription notification will come through for existing jobs in the newly loaded database. Only new jobs generated using the Production Request Editor will work normally with regard to subscriptions.

The scripts used in saving and/or resetting the database are invoked from a UNIX command line prompt. Table 13.5-6 presents (in a condensed format) the steps required to save and/or reset the PDPS database. If you are already familiar with the procedures, you may prefer to use this quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

NOTE: It is important to log in as a user who has "write" permission in the saved_dumps directory (/usr/ecs/<MODE>/CUSTOM/utilities/saved_db/saved_dumps); otherwise it will not be possible to save database contents.

1 Access a terminal window logged in to the Planning/Management Workstation host.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 Set the application environment variables by entering:

setenv ECS_HOME /usr/ecs/

- Application home environment is entered
- When logging in as a system user (e.g., cmshared), the ECS_HOME variable may be set automatically so it may not be necessary to set it manually.

3 In the terminal window, at the command line, enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- **<MODE>** is current mode of operation.
  - TS1 - Science Software Integration and Test (SSI&T)
  - TS2 - New Version Checkout
  - OPS - Normal Operations
- "utilities" is the directory containing the Planning Subsystem utility/start-up scripts.

4 If saving the current database is desired, start the process of saving the database by entering:

EcPlDbSave <Saved_DB_Name>

- **<Saved_DB_Name>** is the desired name for the directory into which the data files will be written.
- If there is not already a directory with the specified **<Saved_DB_Name>**, a **"Making directory..."** message is displayed.
- If there is already a directory with the specified **<Saved_DB_Name>**, the following prompt is displayed:

  Are you sure you wish you overwrite previously saved version of <Saved_DB_Name>? (Y/N)

5 If the prompt **Are you sure you wish you overwrite previously saved version of <Saved_DB_Name>? (Y/N)** appears, enter either **Y** or **N** (as appropriate)

- If **y** is entered, the data are saved in the specified directory, replacing the data that had previously been saved there.
- If **n** is entered, an **"Exiting without saving database..."** message is displayed.

6 If obtaining a listing of saved databases is desired, generate a listing of the saved databases by entering:

EcPlDbList

- A listing of saved databases is displayed.
- The listing is a useful tool to display the options for resetting the database.

7 If resetting the current database is desired, start the database reset process by entering:

EcPlDbReset <Saved_DB_Name>

- **<Saved_DB_Name>** refers to the name of the directory containing the data files to be loaded into the database.
- The data in the database are replaced with the data from the specified directory.
- When the database has been reset a **"Successfully reset the database to <Saved_DB_Name>"** message is displayed.

Table 13.5-6. Save and/or Reset the PDPS Database - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	Set environment variables if necessary	**enter text, press Enter**
3	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**

Table 13.5-6. Save and/or Reset the PDPS Database - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
4	**EcPlDbSave <Saved_DB_Name>** (if applicable)	**enter text, press Enter**
5	Either **Y** or **N** (if applicable)	**enter text, press Enter**
6	**EcPlDbList** (if applicable)	**enter text, press Enter**
7	**EcPlDbReset <Saved_DB_Name>** (if applicable)	**enter text, press Enter**

13.6 Tuning System Parameters

The Production Planner works with the rest of the production team (i.e., Resource Planner and Production Monitors) to tune system parameters. Detailed procedures are included in Chapter 14, Production Processing.

13.7 Troubleshooting Production Planning Problems

Troubleshooting is a process of identifying the source of problems on the basis of observed trouble symptoms. One common source of problems involves connections with other subsystems for the transmission of messages or data. Like many other operational areas in ECS, Planning has interfaces with many other subsystems. Consequently, problems with processing can be traced to either the Planning Subsystem or one of many other ECS subsystems, including (but not necessarily limited to) those in the following list:

- Data Processing Subsystem (DPS).
- Data Server Subsystem (DSS).
- Communications Subsystem (CSS).

Table 13.7-1, below, provides an Activity Checklist for troubleshooting Production Planning problems.

Table 13.7-1. Troubleshooting Production Planning Problems - Activity Checklist (1 of 2)

Order	Role	Task	Section	Complete?
1	Production Planner	Troubleshoot a Production Planning Problem	(P) 13.7.1	
2	Production Planner	Check Connections to Hosts/Servers	(P) 13.7.1.1	
3	Production Planner	Check Log Files	(P) 13.7.1.2	
4	Production Planner	Check Database Connections	(P) 13.7.1.3	
5	Production Planner	Handle a Failure to Generate DPRs	(P) 13.7.2	
6	Production Planner	Check the Production Request Editor ALOG File	(P) 13.7.2.1	

Table 13.7-1. Troubleshooting Production Planning Problems - Activity Checklist (2 of 2)

Order	Role	Task	Section	Complete?
7	Production Planner	Use ISQL to Check Database Tables	(P) 13.7.2.2	
8	Production Planner	Check the PDPS Database for Causes of Failure to Generate DPRs	(P) 13.7.2.3	
9	Production Planner	Check for DPR Explosion Failure Because the Production Request Editor Does Not Query DSS for Data	(P) 13.7.2.4	
10	Production Planner	Check the Production Request Editor Debug File for Evidence of Metadata Queries	(P) 13.7.2.5	
11	Production Planner	Respond to PR or DPR Deletion that Hangs	(P) 13.7.3	
12	Production Planner	Check for Database Deadlocks	(P) 13.7.3.1	
13	Production Planner	Check for Resource Locks in the PDPS Database	(P) 13.7.3.2	
14	Production Planner	Respond to DPR Deletion that Fails	(P) 13.7.4	
15	Production Planner	Handle a DPR Scheduling Failure	(P) 13.7.5	
16	Production Planner	Respond to a "DPR Validation Failed" Error	(P) 13.7.5.1	
17	Production Planner	Respond to an "information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" Error	(P) 13.7.5.2	

Fault Recovery

Each request that crosses a client/server boundary is assigned a system-unique identifier referred to as an RPC ID. (RPC refers to Remote Procedure Call, the mechanism by which requests are submitted from client to server.) The RPC ID facilitates the automatic fault recovery events that occur whenever there is a client or server failure.

- As a request propagates through the system, each associated client/server exchange is assigned a unique RPC ID.
  - The RPC ID for each interaction is derived from the previous RPC ID received by the client for the request. Consequently, all RPC IDs associated with a given request have a common portion that relates the various client/server calls to one another.
  - Given the previous RPC ID, clients consistently reproduce the same RPC ID that was submitted to the server on the subsequent event.
- The concept of reproducible RPC IDs is central to the ECS fault recovery capability.

- When requests are retried from client to server, they are always submitted with the same RPC ID that was used in the original submission of the request, even if either the client or server has crashed between retries.

- The RPC ID is also central to the check-pointing aspect of fault recovery.
  - As requests arrive at fault recovery-enabled servers, they are recorded in a persistent store (typically a database), tagged with the RPC ID, which identifies the request.
  - As the request is serviced, check-pointing state information may be updated in the persistent store, up to and including the completion status of the request.
  - This allows the servers to resume servicing from the last check-pointed state, particularly upon resubmission from a client.

PLANG and PRONG components check-point the following types of information:

- **EcPlSubMgr** - Unprocessed subscription notifications.
- **EcDpPrDeletion** - Interim Delete Requests.
- **EcDpPrEM** - Queued and activated jobs.

Fault Handling

Failure events are classified according to the following three severity levels:

- Fatal error.
  - Returned when a request cannot be serviced, even with operator intervention.
  - For example, if a request is made to distribute data via ftp to a non-existent host, the request is failed with a fatal error.
- Retry error.
  - Potentially recoverable error.
  - Normally, a retry error would be returned to the client only when the server cannot recover from the error automatically.
  - A retry error may require operator assistance during recovery. For example, when using the Production Request Editor GUI, the Production Planner would enter a new name for a production request after being notified that a previously entered name contained too many characters.
- Warning.
  - Provided when operations can proceed without interruption, but an unexpected circumstance was detected.
  - For example, if a client requests removal of a file but the file does not exist, there is no error per se; however, a warning is generated to caution the client that the file to be removed did not exist in the first place.

Transient errors (such as network errors) are always retry errors.

- In general, clients and servers that experience transient retry errors first attempt to recover by retrying the operation automatically.

- One special case of this is “rebinding,” which refers to the process by which a client automatically attempts to re-establish communication with a server in the event communication is disrupted.
  - The disruption may be caused by transient network failure, or by the server crashing or being brought down.
  - In any case, the client automatically attempts to reconnect to the server for a configurable period of time on a client-by-client basis.

ECS processes encountering an error or receiving an error from a server request can either pass the error back to a higher-level client or present it to the operator for operator intervention. The specific fault handling policies for PLANG and PRONG client processes are shown in Table 13.7-2.

Table 13.7-2. PLANG and PRONG Fault Handling Policies

Client Process	Fault Handling Policy
EcPlSubMgr	**Retry errors:** All Subscription processing errors are retried a configurable number of times and for a configurable time period. After the configurable number of times (or time period) the subscription is lost. **Fatal errors:** N/A.
EcPlPREditor_IF **EcPlWb**	**Retry errors:** Since these are GUI applications, errors are reported to the operator and it is the operator’s responsibility to retry the request. **Fatal errors:** Errors are reported to the operator.
EcPlOdMgr	**Retry errors:** Retries errors from the Science Data Server and the Subscription Server. **Fatal errors:** Logs errors and stops current on demand requests.
EcDpPrEM	**Retry errors:** Errors are retried a configurable number of times, then the job is failed and it is up to the Production Monitor to restart the job through AutoSys. **Fatal errors:** A fatal error message is logged.
EcDpPrJobMgmt	**Retry errors:** If a DPR cannot be assigned to a machine or created in AutoSys, it is left in a PENDING state and the assignment is retried after DpPrPendingThreadWaitInterval seconds. **Fatal errors:** N/A.
EcDpPrDeletion	**Retry errors:** No retries are implemented. Status from DSS is not checked. **Fatal errors:** N/A.

Client Crash and Restart

When a client of a PLANG or PRONG server crashes, the server (i.e., EcPlSubMgr, EcDpPrJobMgmt, or EcDpPrDeletion) continues to service the requests that were in process at the time of the client's crash.

When a client restarts in the ECS system, it sends a restart notification to each server with which it interacts.

- Clients notify servers that they have come up either "cold" or "warm."
- Generally, the notification temperature sent to the server matches the temperature at which the client process is restarted.

The default server behavior in response to startup notification from a client is as follows:

- Warm Notification.
  - Outstanding requests for the restarted clients remain available in the persistent store.
  - The outstanding requests may be resubmitted by the client, and are serviced to completion upon resubmission.
  - Associated resources are left allocated until the requests are completed.
- Cold Notification.
  - All outstanding requests for the restarted client are cancelled.
  - If the client resubmits any cancelled request using the same RPC ID (e.g., by pressing the Retry button from an operator GUI), it is failed with a fatal error due to the client cold startup notification.
  - Any resources associated with the cancelled requests are released and reclaimed by the system.

Server Crash and Restart

When a server crashes, clients cannot continue to submit requests for processing.

- Synchronous requests in progress result in a Distributed Computing Environment (DCE) exception being thrown back to the client process, which enters a rebinding failure recovery mode (as previously mentioned).
- Attempts to submit requests while the server is down result in the client blocking until a communication timeout has been reached.
- Although DCE has been replaced by socket-based library calls (i.e., CCS Middleware), the DCE exception code is handled by the CCS Middleware.

When a server restarts, it may perform various resynchronization activities in order to recover from an unexpected termination.

- In the event of a server cold start or cold restart, the server typically cancels all outstanding requests and reclaims all associated resources.
- In general, existing request queues are retained for warm restarts and cleared for cold starts or cold restarts.

- **EcPlSubMgr**-specific activities upon start/restart:
  - **Warm Restart:** Any subscriptions that have not been processed are read from checkpoint file and processed.
  - **Cold Start or Cold Restart:** N/A.
- **EcDpPrJobMgmt**-specific activities upon start/restart:
  - **Warm Restart:** Jobs in AutoSys and jobs waiting in the queue are read from the database; any jobs that are ready are placed into AutoSys from the queue (if there are processing slots available).
  - **Cold Start or Cold Restart:** N/A.
- **EcDpPrDeletion**-specific activities upon start/restart:
  - **Warm Restart:** Interim granules marked for deletion are read from the database and are deleted when time-out occurs.
  - **Cold Start or Cold Restart:** N/A.

Request Resubmission

Upon restarting a crashed client or server, requests are typically resubmitted. If the restarted process was started warm, the fault-recovery capabilities permit the server to resume processing of the request from its last check-pointed state. This prevents needless repetition of potentially time-consuming activities.

- **EcDpPrJobMgmt**- and **EcDpPrDeletion**-specific activities upon resubmission of a request:
  - Requests are submitted synchronously.
  - If the entire request is resubmitted by a client, then only that part of the resubmitted request that has not been completed is reprocessed.

13.7.1 Troubleshoot a Production Planning Problem

1 If it is not possible to log in to the Planning Subsystem host, ask the Operations Controller/System Administrator to verify that the host is “up."

- Examples of Planning Subsystem host names include **e0pls03**, **g0pls01**, and **l0pls02**.

2 If the GUI (e.g., the **Production Request Editor**, the **Production Strategies GUI**, or the **Planning Workbench**) is not displayed when the start-up script has been properly invoked, ensure that the DISPLAY variable was set properly.

- For detailed instructions refer to the applicable procedure.
  - **Launch the Production Request Editor** (Section 13.2.2).
  - **Launch the Production Strategies GUI** (Section 13.3.1).
  - **Launch the Planning Workbench and Planning Timeline GUIs** (Section 13.4.1).

3 If an error message is received indicating that SNS (System Name Server) and/or Resource Model is/are in use using the selected Application ID and if working in a different mode from the person using the selected Application ID, use a different Application ID.

- For detailed instructions refer to the applicable procedure.
  - **Launch the Production Request Editor** (Section 13.2.2).
  - **Launch the Planning Workbench and Planning Timeline GUIs** (Section 13.4.1).

4 If an error message is received indicating that SNS (System Name Server) and/or Resource Model is/are in use using the selected Application ID and if working in the same mode as the person using the selected Application ID, coordinate use of Planning applications with the other user and/or the System Administrator.

5 If an error message associated with the Production Request Editor is received, refer to Table 13.7-3, Production Request Editor User Messages.

- The table is adapted from the corresponding table in 609-EMD-001, *Release 7 Operations Tools Manual for the EMD Project*.

6 If an error message associated with the Production Strategies GUI is received, refer to Table 13.7-4, Production Strategies GUI User Messages.

- The table is adapted from the corresponding table in 609-EMD-001, *Release 7 Operations Tools Manual for the EMD Project*.

7 If an error message associated with the Planning Workbench is received, refer to Table 13.7-5, Planning Workbench User Messages.

- The table is adapted from the corresponding table in 609-EMD-001, *Release 7 Operations Tools Manual for the EMD Project*.

8 If a Production Request fails (DPR generation fails), ensure that it is possible to connect to the necessary hosts and servers.

- For detailed instructions refer to the **Check Connections to Hosts/Servers** procedure (Section 13.7.1.1).

10 If a Production Request fails (DPR generation fails) and if the servers are all "up," perform the **Handle a Failure to Generate DPRs** procedure (Section 13.7.2).

11 If a PR or DPR deletion hangs, ensure that enough time has passed to allow DPR deletion.

- Deleting a DPR can require as much time as creating a DPR.

12 If a PR or DPR deletion hangs and enough time has passed to allow DPR deletion, check for a database lock or resource lock in the PDPS database.

- For detailed instructions refer to the **Respond to PR or DPR Deletion that Hangs** procedure (Section 13.7.3).

13 If DPR deletion fails, ensure that enough time has passed to allow DPR deletion.

- Deleting a DPR can require as much time as creating a DPR.

14 If DPR deletion fails and enough time has passed to allow DPR deletion, check the Deletion Server Debug log (EcDpPrDeletionDebug.log).

- For detailed instructions refer to the **Respond to DPR Deletion that Fails** procedure (Section 13.7.4).

15 If DPR scheduling fails (DPR is not passed to Data Processing), perform the **Handle a DPR Scheduling Failure** procedure (Section 13.7.5).

16 If some other type of problem is encountered, check the log files for error messages.

- Examples of log files include EcPlPREditor.ALOG, EcPlPREditorDebug.log, EcPlWb.ALOG, EcPlWbDebug.log, and EcPlTl.ALOG.
- Log files are located in the /usr/ecs/<MODE>/CUSTOM/logs directory.
- For detailed instructions refer to the **Check Log Files** procedure (Section 13.7.1.2).

17 If the problem cannot be identified and fixed without help within a reasonable period of time, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 13.7-3. Production Request Editor User Messages (1 of 6)

Message Text	Impact	Cause and Corrective Action
Data Processing Request not open, Data Processing Request to be open must be selected.	Unable to open a DPR.	1. **Single-click** on a DPR from the DPR List. 2. Open the DPR using either **File** → **Open** from the pull-down menu or the keyboard **Ctrl-O**. [For detailed instructions refer to the **Review Data Processing Requests** procedure (Section 13.2.7).]
Do you want to delete this DPR <DPRID>?	Requires operator input (confirmation of deletion) before proceeding.	1. **Single-click** on **Yes** to delete the DPR. 2. **Single-click** on **No** to keep the DPR.
Do you want to delete this PR <PRNAME>?	Requires operator input (confirmation of deletion) before proceeding.	1. **Single-click** on **Yes** to delete the PR. 2. **Single-click** on **No** to keep the PR.

Table 13.7-3. Production Request Editor User Messages (2 of 6)

Message Text	Impact	Cause and Corrective Action
DPR Delete Failed.	Selected DPR was not deleted from the PDPS database.	Check the log files (e.g., EcPlPREditor.ALOG and EcPlPREditorDebug.log) in the /usr/ecs/*MODE*/CUSTOM/logs directory for error messages. [For detailed instructions refer to the **Check Log Files** procedure (Section 13.7.1.2).]
Dpr Generation Incomplete for PR <PRNAME>. Do you want to complete Dpr explosion?	Requires operator input before proceeding.	1. **Single-click** on **Yes** to complete the DPR generation. 2. **Single-click** on **No** to display another message that will show how many DPRs have been generated so far.
Environment variable PL_NEW not set.	**Production Request Editor** GUI is not started	1. Submit a request to the Database Administrator to check the value assigned to **PL_NEW** (associated with EcPlPREditor) in the Configuration Registry. 2. If no value is specified for **PL_NEW**, submit a request to the Database Administrator to add the value **PL_NEW = New** for the EcPlPREditor in the Configuration Registry.
Invalid Time is entered or End time is less than Begin Time.	Unable to create Data Processing Requests because the collection/insertion time(s) entered are invalid.	Ensure that the **Begin** and **End** dates/times entered on the **PR Edit** screen are valid.
Must select a PGE for Production Request before saving.	Unable to proceed with Data Processing Request creation until a PGE is entered on the **PR Edit** screen.	1. Open the **PGE** screen (**single-click** the mouse on the **PGE** button of the **PR Edit** screen). 2. Select (highlight) a PGE from the PGE list. 3. **Single-click** on the **OK** button.
<number> Dpr(s) have previously been generated. Are you sure you want to delete this PR?	Requires operator input before proceeding.	1. **Single-click** on **Yes** to delete the PR. 2. **Single-click** on **No** to clear the message.
Please select a Dpr you want to delete from the DPR List.	DPR will not be deleted from the PDPS database until a DPR is selected from DPR List.	1. **Single-click** on a DPR from the DPR List. 2. Delete the DPR using either **Edit** → **Delete** from the pull-down menu or the keyboard **Ctrl-D**. [For detailed instructions refer to the **Delete a Data Processing Request** procedure (Section 13.2.8).]
Please select a DPR you want to open from the DPR list,	DPR cannot be opened on the DPR View page until the DPR is selected from the DPR List.	1. **Single-click** on a DPR from the DPR List. 2. Open the DPR using either **File** → **Open** from the pull-down menu or the keyboard **Ctrl-O**. [For detailed instructions refer to the **Review Data Processing Requests** procedure (Section 13.2.7).]

Table 13.7-3. Production Request Editor User Messages (3 of 6)

Message Text	Impact	Cause and Corrective Action
Please select a PR you want to delete from the PR List.	Cannot delete a Production Request until it is selected from the PR List.	1. **Single-click** on a PR from the PR List. 2. Delete the PR using either **Edit** → **Delete** from the pull-down menu or the keyboard **Ctrl-D**. [For detailed instructions refer to the **Delete a Production Request** procedure (Section 13.2.6).]
Please specify production request to filter by.	Unable to filter Production Request IDs.	**Filter** button acts on the pattern specified in the filter text entry area. Here, the user can filter all the production requests by the specified pattern. 1. If applicable, **single-click** on the appropriate PR type on the **PR Type** option button. 2. **Single-click** on the appropriate PR in the **PR Name** field. 3. **Single-click** on the **Filter** button. [For detailed instructions refer to the **Review Data Processing Requests** procedure (Section 13.2.7).]
PR Failed - Need to include 1st orbit in the Orbit model.	Data Processing Request creation fails because 1st orbit information is incorrect.	1. Enter isql commands for checking the PDPS database PlPgeOrbitModel table to determine whether the orbit information for the first orbit is in the table. [For detailed instructions refer to the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).] 2. If the orbit information for the first orbit is not in the PlPgeOrbitModel table, notify the SSI&T team.
PR Failed - Read failure from PlOrbitModel.	Data Processing Request creation fails because orbit information cannot be read from the PDPS database.	Unable to read the PlPgeOrbitModel table from the PDPS database. 1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 13.7.1.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Production Request <PRNAME> Deleted; <number> DPR(s) deleted from the database.	Production Request(s) and dependent DPR(s) are successfully deleted from the PDPS database.	For information only. The message indicates how many DPRs associated with the PR were deleted.
Production Request Delete Failed.	Selected Production Request(s) is (are) not deleted.	Unable to delete a PR from the database. Check the log files (e.g., EcPlPREditor.ALOG and EcPlPREditorDebug.log) in the /usr/ecs/*MODE*/CUSTOM/logs directory for error messages. [For detailed instructions refer to the **Check Log Files** procedure (Section 13.7.1.2).]

Table 13.7-3. Production Request Editor User Messages (4 of 6)

Message Text	Impact	Cause and Corrective Action
Production Request Explosion into DPR(s) Failed.	Data Processing Request creation failed.	Check the log files (e.g., EcPlPREditor.ALOG and EcPlPREditorDebug.log) in the /usr/ecs/*MODE*/CUSTOM/logs directory for error messages. [For detailed instructions refer to the **Check Log Files** procedure (Section 13.7.1.2).]
Production Request Explosion into DPRs Failed, zero DPRs Generated.	Data Processing Request creation failed.	Check the log files (e.g., EcPlPREditor.ALOG and EcPlPREditorDebug.log) in the /usr/ecs/*MODE*/CUSTOM/logs directory for error messages. [For detailed instructions refer to the **Check Log Files** procedure (Section 13.7.1.2).]
Production Request Explosion into DPRs ok. <number> DPR(s) Generated.	Data Processing Requests successfully created and stored in the PDPS database.	For information only. The message indicates how many DPR(s) were generated during the PR Explosion.
Production Request Explosion into DPR(s) ok. <number> more DPR(s) Generated.	Data Processing Requests successfully created and stored in the PDPS database.	For information only. The message indicates how many more DPRs were generated for the PR.
Production Request not open, Production Request to be open must be selected.	Production Request is not opened on the **PR Edit** screen until a PR is selected from the PR List.	1. **Single-click** on a PR from the PR List. 2. Open the PR using either **File** → **Open** from the pull-down menu or the keyboard **Ctrl-O**. [For detailed instructions refer to the **Edit/Modify a Production Request** procedure (Section 13.2.5).]
Production Request not saved, Production Request already exists.	Data Processing Request creation failed because the Production Request name already exists in the PDPS database.	1. Enter a new name for the production request. 2. Save the PR. [For detailed instructions refer to the **Edit/Modify a Production Request** procedure (Section 13.2.5).]
Production Request not saved, Production Request must have a name before it is saved.	Unable to proceed with Data Processing Request creation because no name has been entered for the Production Request.	1. Enter a new name for the production request. 2. Save the PR. [For detailed instructions refer to the **Create a New Production Request Using the Production Request Editor GUI** procedure (Section 13.2.3).]

Table 13.7-3. Production Request Editor User Messages (5 of 6)

Message Text	Impact	Cause and Corrective Action
Production Request not saved, Production Request must have a name shorter than 20 characters before it is saved	Unable to proceed with Data Processing Request creation because the name chosen for the Production Request exceeds the maximum number of characters allowed.	1. Enter a PR name with fewer than 20 characters. 2. Save the PR. [For detailed instructions refer to the **Edit/Modify a Production Request** procedure (Section 13.2.5).]
SECURITY VIOLATION: no write permission.	Unable to proceed with Data Processing Request creation because Production Request Editor does not have proper “write” permission on the PDPS database.	Operator does not have permission to save a production request. 1. Exit from the Production Request Editor (**File** → **Exit** from the pull-down menu). 2. Log in as a user with "write" permission. [Contact the System Administrator for assistance if necessary.] 3. Launch the Production Request Editor. [For detailed instructions refer to the **Launch the Production Request Editor** procedure (Section 13.2.2).] 4. Create the Production Request. [For detailed instructions refer to the **Create a New Production Request Using the Production Request Editor GUI** procedure (Section 13.2.3).]
Unable to Initialize PlDpr Pool.	**Production Request Editor** GUI is not started (database cannot be accessed).	Unable to read the DPR table from the database. 1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 13.7.1.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Unable to Initialize PlPge Pool.	**Production Request Editor** GUI is not started (database cannot be accessed).	Unable to read the PGE table from the database. 1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 13.7.1.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Unable to Initialize PlProductionRequest Pool.	**Production Request Editor** GUI is not started (database cannot be accessed).	Unable to read the PR table from the database. 1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 13.7.1.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 13.7-3. Production Request Editor User Messages (6 of 6)

Message Text	Impact	Cause and Corrective Action
Write to Database of Production Request Failed.	Data Processing Request creation fails because the Production Request information cannot be saved in the PDPS database.	Unable to write the data to the database. 1. Check the database connections. [For detailed instructions refer to the **Check Database Connections** procedure (Section 13.7.1.3).] 2. If the problem recurs, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.
Zero DPR Generated. Production Request Not Saved.	Data Processing Request creation failed.	The proposed Production Request might result in the creation of a duplicate DPR. 1. Compare the characteristics of the proposed PR and existing DPRs to determine whether the Production Request would result in the creation of a duplicate DPR. [For detailed instructions refer to the **Review Data Processing Requests** procedure (Section 13.2.7).] 2. Ensure that it is possible to connect to the necessary hosts and servers (listed in Table 13.7-7). [For detailed instructions refer to the **Check Connections To Hosts/Servers** procedure (Section 13.7.1.1).] 3. If hosts/servers are all "up," perform the **Handle a Failure to Generate DPRs** procedure (Section 13.7.2). 4. Retry generating DPRs by resaving the Production Request. [For detailed instructions refer to the **Edit/Modify a Production Request** procedure (Section 13.2.5).] 5. Check the log files (e.g., EcPlPREditor.ALOG and EcPlPREditorDebug.log) in the /usr/ecs/*MODE*/CUSTOM/logs directory for error messages. [For detailed instructions refer to the **Check Log Files** procedure (Section 13.7.1.2).]

Table 13.7-4. Production Strategies GUI User Messages (1 of 2)

Message Text	Impact	Cause and Corrective Action
Invalid StrategyID. Please enter a StrategyID.	Unable to save the Production Strategy.	The strategy ID text field is blank. Enter a valid Strategy ID. [For detailed instructions refer to the **Define or Modify a Production Strategy** procedure (Section 13.3.2).]
No StrategyID selected. Please select one.	Unable to open or delete a Strategy	No strategy is selected from the Production Strategy list. **Single-click** on a Strategy ID from the list.
This StrategyID already exists.	Unable to save Production Strategy.	The Strategy ID entered already exists. Enter a new name for the Strategy ID.

Table 13.7-4. Production Strategies GUI User Messages (2 of 2)

Message Text	Impact	Cause and Corrective Action
Total Weight must be 100 – Normalize weights.	Unable to save Production Strategy.	The combined weights for the different "Types" exceed 100. **Single-click** on the **Normalize** push button.
Value out of range (0-10).	Unable to save the Production Strategy.	Default value entered for the selected "Type" is out of range and is reset to this strategy's default value for this "Type". Enter a value in the range of 0 – 10.
Value out of range (0-100).	Unable to save the Production Strategy.	Weight for the selected "Type" is out of range and is reset to this strategy's default weight for this "Type". Enter a value in the range of 0 – 100.

Table 13.7-5. Planning Workbench User Messages

Message Text	Impact	Cause and Corrective Action
The following dpr jobs were failed when you activated the plan <plan name> <list of failed dprs, one per line> Do you want to recover the failed dpr jobs?	Planning Workbench detects that Job Management has not received the DPRs correctly. The operator needs to be aware that there is (are) failed job(s).	**Single-click** on **Recover**; the failed jobs will be resubmitted.
You cannot reschedule with the current active plan. Do you want to create new plan to schedule production request(s) and to reactivate the plan?	Cannot replan under the same plan name.	1. **Single-click** on **Yes** to create a new plan that can be used for replanning. 2. **Single-click** on **No** and use an existing plan to activate and replan.
You have already submitted the plan, <plan name> and this plan is currently active. Do you want to create a new plan?	If the operator has already submitted this plan for activation, the Planning Workbench will not allow it to be submitted twice.	1. **Single-click Yes** to create a new plan that can be used for replanning. 2. **Single-click No** and use an existing plan to activate and replan.
You need to save the current plan before you activate the plan!!!	Insures consistency between the database active plan and what actually gets activated.	**Single-click** on **Yes** to save the plan. The next time activation is attempted there will be no error message.

13.7.1.1 Check Connections to Hosts/Servers

The procedure to **Check Connections to Hosts/Servers** is a part of the **Troubleshoot a Production Planning Problem** procedure (Section 13.7.1). Table 13.7-6 presents (in a condensed format) the steps required to check connections to hosts/servers. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the detailed procedures that follow.

1 Access a terminal window logged in to the Planning/Management Workstation.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- Most other ECS hosts are acceptable for checking connections.
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/utilities

- Change directory to the directory containing the utility scripts.

3 At the command line prompt enter:

EcCsIdPingServers <MODE>

- The following type of response is displayed (only a few representative lines are shown):

  /usr/ecs/TS2/CUSTOM/bin/CSS/Sweeper -nsh x0icg01 -nsp 18202
  FoSwSweeper application started...
  We made a connection with EntryId =x0ins01:38709:23057 --- EcSrTransportSubServer
  We made a connection with EntryId =x0ins01:38712:23057 --- EcSrTransportSubEventServer
  We made a connection with EntryId =x0acs06:33379:17033 --- DsShQuitIDL
  We made a connection with EntryId =x0wkg01:11959:41838305 --- EcDsHdfEosServer_3_G3
  […]

4 Observe the results displayed on the screen to determine whether connections can be made with the necessary hosts and servers.

- The necessary hosts and servers are listed in Table 13.7-7, Hosts, Servers, Clients and Other Software Relevant to Production Planning and Processing.

5 If pinging the servers (Step 3) indicated a problem with any connection, ping the servers again (at the command line prompt enter: **EcCsIdPingServers <MODE>**).

6 Observe the results displayed on the screen to determine whether connections can be made with the necessary hosts and servers.

7 If it is not possible to connect to any needed host(s)/server(s), notify the Operations Controller/System Administrator to check the hosts/servers and bring them back up if necessary.

8 Return to the procedure that recommended checking connections to hosts.

Table 13.7-6. Check Connections to Hosts/Servers - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/utilities**	**enter text, press Enter**
3	**EcCsIdPingServers <MODE>**	**enter text, press Enter**
4	**EcCsIdPingServers <MODE>** [again]	**enter text, press Enter**
5	Identify hosts and servers with which connections cannot be made	**read text**
6	Notify the Operations Controller/System Administrator to bring hosts/servers back up (if applicable)	**contact Operations Controller**
7	Return to the procedure that recommended checking connections to hosts	

Table 13.7-7. Hosts, Servers, Clients and Other Software Relevant to Production Planning and Processing (1 of 2)

HOST	SERVER/CLIENT/OTHER SOFTWARE
Planning/Management Workstation	Production Request Editor (EcPlPREditor) Planning Workbench GUI (EcPlWb) Production Strategies GUI (EcPlProdStrat) Production Planning Master Timeline (EcPlTl) Message Handler (EcPlMsh) System Name Server (EcPlSns) Resource Model (EcPlRm)

Table 13.7-7. Hosts, Servers, Clients and Other Software Relevant to Production Planning and Processing (2 of 2)

HOST	SERVER/CLIENT/OTHER SOFTWARE
Queuing Server (e.g., x0sps02)	Job Management Server (EcDpPrJobMgmt) Deletion Server (EcDpPrDeletion) Execution Management (EcDpPrEM) AutoSys Event Processor (event_demon) AutoSys Event Server (Sybase server) (e.g., x0sps02_srvr) On-Demand Manager (EcPlOdMgr) Subscription Manager (EcPlSubMgr) PDPS database Sybase server (e.g., x0sps02_srvr)
Science Processor (e.g., x0spg11)	PGE Management (EcDpPrRunPGE) Resource Usage (EcDpPrRusage) PGE
Access/Process Coordinators (APC) Server (e.g., x0acg01)	Archive Server (EcDsStArchiveServer) FTP Server (EcDsStFtpServer) Cache Manager Server (EcDsStCacheManagerServer) Staging Disk Server (EcDsStStagingDiskServer) Pull Monitor Server (EcDsStPullMonitorServer)
Ingest Server (e.g., x0icg01)	Name Server (EcCsIdNameServer) Registry Server (EcCsRegistry)
Sun internal server (e.g., x0acs06)	Science Data Server (EcDsScienceDataServer) Data Dictionary (EcDmDictServer) Subscription Server (EcSbSubServer) Event Server (EcSbEventServer)

NOTE: Depending on the installation, software may be loaded on hosts other than the examples provided.

13.7.1.2 Check Log Files

Log files can provide indications of the following types of problems:

- Communication problems.
- Database problems.
- Lack of disk space.

Table 13.7-8 presents (in a condensed format) the steps required to check log files. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Access a terminal window logged in to the appropriate host.

- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- **<MODE>** is current mode of operation.
  - TS1 - Science Software Integration and Test (SSI&T)
  - TS2 - New Version Checkout
  - OPS - Normal Operations
- "logs" is the directory containing Production Planning log files (e.g., EcPlPREditor.ALOG, EcPlPREditorDebug.log, EcPlWb.ALOG, EcPlWbDebug.log, and EcPlTl.ALOG).

3 At the command line prompt enter:

pg <file name>

- **<file name>** refers to the Production Planning log file to be reviewed (e.g., EcPlPREditor.ALOG, EcPlPREditorDebug.log, EcPlWb.ALOG, EcPlWbDebug.log, or EcPlTl.ALOG).
- The first page of the log file is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **vi**, **view**) can be used to review the log file.

4 Review the log file to identify problems that have occurred.

- To exit from **pg** at the **:** prompt enter:

  q

  - The command line prompt is displayed.

5 Respond to problems as follows:

- Production Planning-related problems.
  - Perform the appropriate procedure(s) from Table 13.7-1, Troubleshooting Production Planning Problems.
- Communication problems.
  - Notify the Operations Controller/System Administrator of suspected communication problems.
- Database problems.
  - Verify that relevant database servers are running.
  - Check for lack of (or corruption of) data in the database using either a database browser or interactive structured query language (isql) commands.
  - Notify the Database Administrator of suspected database problems.
- Lack of disk space.
  - Remove unnecessary files.
  - Notify the Operations Controller/System Administrator of recurring disk space problems.

Table 13.7-8. Check Log Files - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/logs**	**enter text, press Enter**
3	**pg <file name>**	**enter text, press Enter**
4	Identify problems indicated in the log file	**read text**
5	Respond to problems as necessary	

13.7.1.3 Check Database Connections

If applications (including the GUIs) are unable to connect to the database, data cannot be retrieved or (in the case of the GUIs) displayed. Consequently, if a GUI does not display data or if the display does not refresh, checking the database connections is a logical step in trying to isolate the problem.

Table 13.7-9 presents (in a condensed format) the steps required to check database connections. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Submit a request to the Database Administrator to identify the values for parameters associated with the appropriate application.

- The following parameters should be requested:
  - **DBName.**
  - **DBServer.**
  - **DBMaxConnections.**
- The preceding parameters are associated with the following applications:
  - EcPlPREditor.
  - EcPlProdStrat.
  - EcPlRm.
  - EcPlSubsEdit.
  - EcPlTl.
  - EcPlWb.

2 Access a terminal window logged in to the Queuing Server host.

- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03**.
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

3 At the command line prompt enter:

isql –U <User ID> -S <Database Server>

- **<User ID>** is the database user's identification; e.g., **pdps_role**.
- **<Database Server>** is the database server; e.g., **x0sps02_srvr**.
- For example:

  isql –U pdps_role –S x0sps02_srvr

4 At the **Password:** prompt enter:

<database password>

- **<database password>** is the password for logging in to the database using the specified **<User ID>**.
- A **1>** prompt is displayed, indicating that a connection has been made with the database.

5 At the **1>** prompt enter:

sp_who

6 At the **2>** prompt enter:

go

- A listing of connections to the database is displayed.
- The listing includes data in the following columns::
  - **spid.**
  - **status.**
  - **loginame.**
  - **hostname.**
  - **blk.**
  - **dbname.**
  - **cmd.**

7 At the **1>** prompt enter:

sp_configure "user connections"

8 At the **2>** prompt enter:

go

- A listing of connections to the database is displayed.
- The listing includes the following types of data:
  - **Parameter Name** (i.e., number of user connections).
  - **Default.**

- **Memory Used.**
- **Config Value.**
- **Run Value.**

9 At the **1>** prompt enter:

quit

- The connection with the database is discontinued.

10 Compare the number of actual connections (results of **sp_who**) with the number of connections for which the database has been configured (results of **sp_configure "user connections"**).

11 If the number of actual connections is very close to the number of connections for which the database has been configured, notify the Database Administrator of the fact.

12 If the number of actual connections is **not** very close to the number of connections for which the database has been configured, compare the number of actual connections with the value for DBMaxConnections that the Database Administrator specified (Step 1).

13 If the number of actual connections is very close to the value for DBMaxConnections, notify the Database Administrator of the fact.

- It may be advisable to increase the value assigned to the DBMaxConnections parameter in the Configuration Registry.

Table 13.7-9. Check Database Connections - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Identify the values for database parameters associated with the appropriate Production Planning application	**contact Database Administrator**
2	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
3	**isql –U <User ID> -S <Database Server>**	**enter text, press Enter**
4	**<database password>**	**enter text, press Enter**
5	**sp_who**	**enter text, press Enter**
6	**go**	**enter text, press Enter**
7	**sp_configure "user connections"**	**enter text, press Enter**
8	**go**	**enter text, press Enter**
9	**quit**	**enter text, press Enter**

Table 13.7-9. Check Database Connections - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
10	Compare the number of actual connections with the number of connections for which the database has been configured	**read text**
11	Notify the Database Administrator of the results	**contact Database Administrator**

13.7.2 Handle a Failure to Generate DPRs

There are several possible reasons for the Planning Subsystem to fail to generate DPRs. A failure to generate DPRs could be a normal consequence of one of the production rules. For example, if the Closest Granule production rule applies, there may be times when no acceptable input granules are found. However, a failure to generate DPRs is most likely due to one of the following problems:

- Error creating a subscription for input data.
- **Production Request Editor** does not query DSS.
- Too many granules meet the criteria for input granules for a particular DPR.

A failure due to “too many granules” is caused when too many granules meet the criteria for input granules for a particular DPR.

- At PGE registration, the number of granules expected for each input ESDT is defined.
  - The definition includes the minimum number and the maximum number of granules expected.
- If the number of granules found is not between the minimum and maximum number, the request fails.
  - The failure occurs either when trying to save the production request or when the PCF file is generated (AutoSys pre-processing step).

The following procedures may be involved in investigating a failure to generate DPRs:

- Check the Production Request Editor ALOG File.
- Use ISQL to Check Database Tables.
- Check the PDPS Database for Causes of Failure to Generate DPRs.
- Check for DPR Explosion Failure Because the Production Request Editor Does Not Query DSS for Data.
- Check the Production Request Editor Debug File for Evidence of Metadata Queries.

13.7.2.1 Check the Production Request Editor ALOG File

Table 13.7-10 presents (in a condensed format) the steps required to check the **Production Request Editor** ALOG file. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Access a terminal window logged in to the Planning/Management Workstation.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- Change directory to the directory containing the planning log files (e.g., EcPlPREditor.ALOG).

3 At the command line prompt enter:

pg <file name>

- **<file name>** refers to the planning log file to be reviewed (e.g., EcPlPREditor.ALOG).
- The first page of the log file is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **vi**, **view**) can be used to review the log file.

4 Review the log file to determine whether the log contains an error message that indicates why the DPR generation failed.

- For example:

  Msg: PlPge::GetInputForDpr - Extr input to process DPR MoPGE01#2007081600OPS, for data type id MOD000#001, with logical id 599001. PlDataTypeReq has a scienceGroup of for this datatype. Expected 2 max inputs, but got 3. Priority : 2 Time : 07/09/99 17:10:52

  – In the example the **Production Request Editor** queried the PDPS database for granules that would satisfy the data needs for the DPR and found three granules instead of the two it expected (i.e., “Expected 2 max inputs, but got 3”).
  – Consequently, the Production Request failed due to “too many granules.”
- To exit from **pg** at the **:** prompt enter:

  q

  – The command line prompt is displayed.

5 If the log does **not** contain an error message that indicates why the DPR generation failed, perform the **Check the PDPS Database for Causes of Failure to Generate DPRs** procedure (Section 13.7.2.3).

6 If there is an error in the log file indicating that too many granules meet the criteria for input granules for a particular DPR, go to the **Check the PDPS Database for Causes of Failure to Generate DPRs** procedure (Section 13.7.2.3).

7 If there is an error in the log file indicating that an entry could not be found for a specific ESDT in the appropriate database, notify the Science Data Specialist or whoever else installs ESDTs to verify that the corresponding ESDT(s) has/have been properly installed.

- If no entry can be found for the ESDT in question it indicates either of the following conditions:
  - The ESDT was not installed.
  - The ESDT was not installed properly.

8 If there is an error in the log file indicating a subscription error, notify the Science Data Specialist or whoever else installs ESDTs to verify that the corresponding ESDT(s) has/have been properly installed.

- If subscription errors exist in the log, it is most likely that the ESDT in question (for the input) was either not installed or improperly installed.
- Notify the Science Data Specialist or whoever else installs ESDTs to verify that the corresponding ESDT(s) has/have been properly installed.

9 If there is an error in the log file indicating communication problems, notify the Operations Controller/System Administrator of the suspected communication problems.

10 Access a terminal window logged in to the Queuing Server host.

- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03**.
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

11 If there is an error in the log file indicating database problems, at the command line prompt (for the Queuing Server) enter:

ps –ef | grep sybase

- A listing of processes containing the string "sybase" is displayed:
- If the PDPS database server is running, the following type of message would be displayed:

  /usr/ecs/OPS/COTS/sybase/bin/dataserver -d/dev/rdsk/c0t2d0s4 -sx0sps02_srvr -e/

- If the PDPS database server were **not** running, the following type of message only would be displayed:

  cmshared 15516 15405 0 14:20:33 pts/6 0:00 grep sybase

12 If the database servers are **not** running, notify the Database Administrator of the problem.

13 If there is an error in the log file indicating database problems and the database servers are running, go to the **Check the PDPS Database for Causes of Failure to Generate DPRs** procedure (Section 13.7.2.3).

- Check for lack of (or corruption of) data in the database.

14 If there is an error in the log file indicating lack of disk space, access a terminal window logged in to the Planning/Management Workstation.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

15 If there is an error in the log file indicating lack of disk space, at the command line prompt (for the Planning/Management Workstation) enter:

cd /<path>

- **<path>** indicates the directory path to a directory (on the disk that is full) from which files can be deleted.

16 If there is an error in the log file indicating lack of disk space, at the command line prompt (for the Planning/Management Workstation) enter:

rm <file name 1> … <file name x>

- **<file name 1> … <file name x>** represent unnecessary files to be removed.
- Use wild cards if appropriate.
  - For example:

    rm *.hdf

    would prompt the system to remove all files with a suffix of .hdf.

17 If removing files from a full disk, respond appropriately to system prompts.

- For example:

  rm: remove trashfile.rs (yes/no)? y

18 If there is a recurring disk space problem, notify the Operations Controller/System Administrator of the problem.

19 If the log contains an error message which indicates that a configuration item in another subsystem may be the source of the problems, consult with the relevant technician (e.g., Distribution Technician) to request assistance in isolating the problem.

20 When the problem has been corrected, use the **Create a New Production Request Using the Production Request Editor GUI** procedure (Section 13.2.3) to re-create the Production Request that led to discovery of the problem.

Table 13.7-10. Check the Production Request Editor ALOG File - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/logs**	**enter text, press Enter**
3	**pg <file name>**	**enter text, press Enter**
4	Review the log file	**read text**
5	Check the PDPS database for causes of failure to generate DPRs (if applicable)	Use procedure in Section 13.7.2.3
6	Notify the Science Data Specialist to verify that the corresponding ESDT(s) has/have been properly installed (If applicable)	**contact Science Data Specialist**
7	Notify the Operations Controller/System Administrator of suspected communication problems (if applicable)	**contact Operations Controller**
8	UNIX window (Queuing Server host)	**single-click** or use procedure in Section 13.2.1
9	**ps –ef | grep sybase**	**enter text, press Enter**
10	Notify the Database Administrator if the database servers are not running	**contact Database Administrator**
11	Check the PDPS database for causes of failure to generate DPRs (if applicable)	Use procedure in Section 13.7.2.3
12	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
13	**cd /<path>** [for cleaning up disk]	**enter text, press Enter**
14	**rm <file name 1> … <file name x>**	**enter text, press Enter**
15	Consult with other technicians to isolate the problem (as needed)	**contact technician(s)**
16	Create a new Production Request when the problem has been corrected	Use procedure in Section 13.2.3

13.7.2.2   Use ISQL to Check Database Tables

The PDPS database is the repository of data concerning PGEs, Production Requests, Data Processing Requests, Production Strategies, Production Plans and other production-related data. The Subscription Server (SUBSRV) database contains data concerning subscriptions.

The data stored in the database can be checked using either a database browser or isql commands. The procedure in this section describes how to check the tables using isql commands.

Table 13.7-11 presents (in a condensed format) the steps required to use isql to check database tables. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Access a terminal window logged in to the appropriate host.

- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03**.
- Examples of Subscription Server host (Sun internal server host) names include **e0acs06**, **g0acs06**, and **l0acs06**.
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 At the command line prompt enter:

isql –U <user ID> -S <database server>

- **<user ID>** is the database user's identification; e.g., **pdps_role**.
- **<database server>** is the database server; e.g., **x0sps02_srvr**.
- For example:

  isql –U pdps_role –S x0sps02_srvr

3 At the **Password:** prompt enter:

<database password>

- **<database password>** is the password for logging in to the database using the specified **<user ID>**.
- A **1>** prompt is displayed, indicating that a connection has been made with the database.

4 At the **1>** prompt enter:

use <database name>

- The **<database name>** is likely to be one of the following names:
  - **pdps** [OPS mode].
  - **pdps_TS1** [TS1 mode].
  - **pdps_TS2** [TS2 mode].

5 At the **2>** prompt enter:

go

6 At the **1>** prompt enter:

select * from <table name>

- Alternatively, enter:

  select <column name> from <table name>

- Another alternative:

  select <column name1>,<column name2>[,<column name3>,…] from <table name>

7 At the **2>** prompt enter:

go

- Table contents are displayed.
  - If * was specified, all entries in the table are displayed.
  - If specific column names were entered, the data associated with those columns only are displayed.

8 To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.

Table 13.7-11. Use ISQL to Check Database Tables - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (appropriate host)	**single-click** or use procedure in Section 13.2.1
2	**isql –U <user ID> -S <database server>**	**enter text, press Enter**
3	**<database password>**	**enter text, press Enter**
4	**use <database name>**	**enter text, press Enter**
5	**go**	**enter text, press Enter**
6	**select * from <table name>**	**enter text, press Enter**
7	**go**	**enter text, press Enter**
8	**quit**	**enter text, press Enter**

13.7.2.3 Check the PDPS Database for Causes of Failure to Generate DPRs

The PDPS database is a useful resource for troubleshooting a failure to generate DPRs. Certain values must be entered in tables in the PDPS database during the DPR generation process in order for the DPRs to be successfully generated.

- The subscriptionFlag values (PlDataTypeMaster table) for all the data types (ESDTs) needed for the PGE are set.

- The Science Data Server UR is entered in the dataServUrString column (PlDataTypeMaster table) for all data types (ESDTs) needed for the PGE.
- If the PGE specified in the Production Request requires static files, the URs for the static files are included in PlDataGranuleShort table (universalReference column).

Table 13.7-12 presents (in a condensed format) the steps required to check the PDPS database for causes of failure to generate DPRs. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Log in to the PDPS database.

- Database log-in is described in Steps 1 through 5 of the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).

2 At the **1>** prompt enter:

select dataTypeId,subscriptionFlag from PlDataTypeMaster

- Prepare a request to view PlDataTypeMaster table subscriptionFlag column values for all data types (dataTypeId column).

3 At the **2>** prompt enter:

go

- Contents of specified columns of the **PlDataTypeMaster** table are displayed.
- During DPR generation, the DPR executable should turn the subscriptionFlag for all the data types (ESDTs) needed for the PGE from zero to non-zero.
- If the subscriptionFlag values for the data types did not turn to non-zero, subscription trouble is indicated.

4 If the subscriptionFlag value for any ESDT needed for the PGE is **zero**, make a note of the fact for subsequent reporting of the problem.

5 At the **1>** prompt enter:

select dataTypeId,dataServUrString from PlDataTypeMaster

- Prepare a request to view PlDataTypeMaster table dataServUrString column values for all data types.

6 At the **2>** prompt enter:

go

- Contents of specified columns of the **PlDataTypeMaster** table are displayed.
- During DPR generation, the DPR executable should turn the dataServUrString for all the ESDTs needed for the PGE from “NULL” to the UR value for Science Data Server (e.g., UR:15:DsShSciServerUR:13:[MDC:DSSDSRV]).

7 If the dataServUrString value for any ESDTs needed for the PGE does not have the UR value for the Science Data server (e.g., is NULL), make a note of the fact for subsequent reporting of the problem.

8 If the PGE specified in the Production Request requires static files, at the **1>** prompt enter:

select dataTypeId,universalReference from PlDataGranuleShort

- Prepare a request to view PlDataGranuleShort table universalReference column for the URs of the necessary static files.
- If there is data on many data types in the PlDataGranuleShort table, it may be advisable to limit the search, for example:

  1> select dataTypeId,universalReference from PlDataGranuleShort where dataTypeId = "AM1ATTNF#001"

  – In the example **AM1ATTNF#001** is the data type being checked.

9 At the **2>** prompt enter:

go

- Contents of specified columns of the **PlDataGranuleShort** table are displayed.
- If the PGE requires static files, the URs for the static files must be included in the PlDataGranuleShort table in order for a DPR to be successfully generated.
- For dynamic granules, the corresponding UR values may become available during DPR generation; however, if the dynamic granules' URs do not become available during DPR generation, there is no effect on DPR generation.

10 If the PGE requires static files, and the URs for the static files are **not** in the PlDataGranuleShort table, make a note of the fact for subsequent reporting of the problem.

11 To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.

12 If any problems were noted in the PDPS database, report the problem(s) to the SSI&T team and/or call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

- There may be a problem with PGE registration (especially if no other PGEs are affected) or other problem associated with the SSI&T process.

Table 13.7-12. Check the PDPS Database for Causes of Failure to Generate DPRs - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Log in to the PDPS database	Use procedure in Section 13.7.2.2
2	**select dataTypeId,subscriptionFlag from PlDataTypeMaster**	**enter text, press Enter**
3	**go**	**enter text, press Enter**
4	**select dataTypeId,dataServUrString from PlDataTypeMaster**	**enter text, press Enter**
5	**go**	**enter text, press Enter**
6	**select dataTypeId,universalReference from PlDataGranuleShort**	**enter text, press Enter**
7	**go**	**enter text, press Enter**
8	**quit**	**enter text, press Enter**
9	Report all problems to the SSI&T team and/or call the help desk and submit a trouble ticket	**contact SSI&T team**

13.7.2.4 Check for DPR Explosion Failure Because the Production Request Editor Does Not Query DSS for Data

If the Production Request Editor reports a failure creating a DPR that is subject to the Metadata Query production rule, the process for responding to the problem consists of the following activities:

- Check the "debug" log for evidence that the Science Data Server was queried for metadata values.
- If there is no message in the debug log indicating that the Science Data Server was queried for metadata values, the SSI&T team should check for an error in the appropriate PGE science metadata ODL file.

13.7.2.5 Check the Production Request Editor Debug File for Evidence of Metadata Queries

Table 13.7-13 presents (in a condensed format) the steps required to check the Production Request Editor debug file for evidence of metadata queries. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Access a terminal window logged in to the Planning/Management Workstation.

- Examples of Planning/Management Workstation host names include **e0pls03**, **g0pls01,** and **l0pls02.**

- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- Change directory to the directory containing the planning log files (e.g., EcPlPREditorDebug.log).

3 At the command line prompt enter:

pg <file name>

- **<file name>** refers to the planning log file to be reviewed (e.g., EcPlPREditorDebug.log).
- The first page of the log file is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **vi**, **view**) can be used to review the log file.

4 Review the log file to determine whether the log contains a message that indicates why the DPR generation failed.

- When the Production Request Editor queries the Science Data Server to search for the PGE's input(s), the query should include the metadata value(s) desired by the PGE.
- Note that events are recorded in the debug log in chronological order and are preceded by a time and date stamp.
- If there is no message which indicates that the Science Data Server was queried for the metadata value(s) desired by the PGE, there might be an error in the PGE Metadata ODL File for the PGE.
- To exit from **pg** at the **:** prompt enter:

  q

  – The command line prompt is displayed.

5 If there was no message in the debug log indicating that the Science Data Server was queried for metadata values, submit a request to the SSI&T team to check for an error in the appropriate PGE science metadata ODL file and re-register the PGE.

- After the SSI&T team has edited the ODL file and re-registered the PGE it should be possible to successfully create DPRs for the PGE.

Table 13.7-13. Check the Production Request Editor Debug File for Evidence of Metadata Queries - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	UNIX window (Planning/Management Workstation)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/logs**	**enter text, press Enter**

Table 13.7-13. Check the Production Request Editor Debug File for Evidence of Metadata Queries - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
3	**pg <file name>**	**enter text, press Enter**
4	Review the log file	**read text**
5	Ask the SSI&T team to check for errors in the PGE science metadata ODL file (if there was no metadata query message in the debug log)	**contact SSI&T team**

13.7.3 Respond to PR or DPR Deletion that Hangs

When deleting DPRs or PRs from the Production Request Editor, the Job Management Server and Deletion Server (both in DPS) are called to clean up all PDPS database tables associated with the DPR or PR. It is possible that any failure or "hung" condition can be attributed to the Production Request Editor itself or the Deletion or Job Management Server. If the servers are functioning properly, there are two other possible causes:

- The database is locked.
- There may be resource locks in the PDPS database.

Accordingly, the following procedures may be involved in responding to PR or DPR deletion that hangs:

- Check for Database Deadlocks.
- Check for Resource Locks in the PDPS Database.

13.7.3.1 Check for Database Deadlocks

A deadlock occurs when a database transaction locks a record that another transaction needs and the second transaction locks the record that first transaction needs. Each program must wait until the other completes. However, neither can complete (because each is waiting for the other) so both end up waiting indefinitely.

Table 13.7-14 presents (in a condensed format) the steps required to check for database deadlocks. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Log in to the PDPS database.

- Database log-in is described in Steps 1 through 5 of the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).

2 At the **1>** prompt enter:

sp_lock

3 At the **2>** prompt enter:

go

- Results displayed include the following features:
  - **spid** column shows the process id. The database user that owns a process can be determined using the **sp_who** isql command.
  - **locktype** column indicates a problem if the entry starts with "Ex_" (exclusive).
  - **table_id** column identifies the table that the corresponding spid has locked. The name of the table can be determined using the **select** command [i.e., **select object_name (table_id)**].

4 At the **1>** prompt enter:

select object_name (<table id>)

- For example, to check the exclusive locks related to spid 24, table ID 197575742, enter:

  1> select object_name (197575742)

5 At the **2>** prompt enter:

go

- The object name is displayed (e.g., PlDprData).

6 At the **1>** prompt enter:

sp_who

7 At the **2>** prompt enter:

go

- A listing of connections to the database is displayed.
- The listing includes data in the following columns:
  - **spid.**
  - **status.**
  - **loginame.**
  - **hostname.**
  - **blk.**
  - **dbname.**
  - **cmd.**

8 Analyze the results of the request.

- The **blk** column shows the spid of the process that is doing the blocking.
- The **cmd** column shows the command that the blocked process is trying to complete.

9 To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.

10 If there is a deadlock in the database, ask the Operations Controller to bounce the server that is causing the deadlock.

11 If there is no deadlock, perform the **Check for Resource Locks in the PDPS Database** procedure (Section 13.7.3.1).

Table 13.7-14. Check for Database Deadlocks - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Log in to the PDPS database	Use procedure in Section 13.7.2.2
2	**sp_lock**	**enter text, press Enter**
3	**go**	**enter text, press Enter**
4	**select object_name (<table id>)**	**enter text, press Enter**
5	**go**	**enter text, press Enter**
6	**sp_who**	**enter text, press Enter**
7	**go**	**enter text, press Enter**
8	Analyze the results of the request	**read text**
9	**quit**	**enter text, press Enter**
10	Ask the Operations Controller to bounce any server that is causing a deadlock (if applicable)	**contact Operations Controller**
11	Check for resource locks in the PDPS database (if there is no deadlock)	Use procedure in Section 13.7.3.1

13.7.3.1 Check for Resource Locks in the PDPS Database

Resource locks used to occur if there was an attempt to delete DPRs/PRs while their corresponding jobs were still running in AutoSys or jobs had been explicitly killed before the DPRs/PRs were deleted. However, resource locking has been removed for all Resource Management calls (e.g., for allocating CPUs and disk space). The locks have been replaced with the following features:

- Sybase stored procedures that use transactions.
- Database triggers.

Resource locking is still used for disk space reclamation. Momentary system interruptions occur during the process of disk space reclamation. The interruptions may happen several times a day. The system may look like it is "hung" during such periods. The procedure that follows should be performed to verify that disk space reclamation is proceeding normally:

Although the procedure for checking for resource locks in the PDPS database includes the use of isql commands, an acceptable alternative is to use a database browser to check the contents of the DpPrResourceLock table.

Table 13.7-15 presents (in a condensed format) the steps required to check for resource locks in the PDPS database. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Log in to the PDPS database.

- Database log-in is described in Steps 1 through 5 of the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).

2 At the **1>** prompt enter:

select * from DpPrResourceLock

- Prepare a request to view the contents of the DpPrResourceLock table.

3 At the **2>** prompt enter:

go

- The contents of the **DpPrResourceLock** table are displayed.
- The listing includes data in the following columns:
  - **jobId.**
  - **priority.**
  - **ecsUnit.**
  - **attempts.**
  - **state.**
  - **pid.**
  - **queuePosition.**

4 Analyze the results of the request.

- A jobId with a state <> 0 would indicate a resource lock.
  - If there are entries in the **DpPrResourceLock** table and there are no other jobs running in AutoSys, all entries in the table need to be deleted before the DPR/PR deletion can complete.
  - If other jobs (DPRs) are currently being executed in AutoSys and the other jobs should not be deleted, the entries in the table that need to be deleted are those related to the job to be deleted only. The entries concerning the other (running) jobs must be left in the table.
- If there is no evidence of a resource lock, go to Step 8.

5 If all entries in the DpPrResourceLock table are to be deleted, at the **1>** prompt enter:

delete DpPrResourceLock

- Go to Step 7.

6 If some (but not all) entries in the DpPrResourceLock table are to be deleted, at the **1>** prompt enter:

delete DpPrResourceLock where jobId like "<job Id>"

- **<job Id>** specifies the job whose entries are to be deleted.

7 At the **2>** prompt enter:

go

- Entries in the **DpPrResourceLock** table are deleted.
- The DPR/PR deletion that was delayed by the resource lock should go to completion.

8 To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.
- If entries were deleted from the DpPrResourceLock table the procedure is finished; otherwise, continue with Step 9.

9 Access a terminal window logged in to the Queuing Server.

- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03**.
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

10 At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- Change directory to the directory containing the data processing log files (e.g., EcDpPrJobMgmt.ALOG, EcDpPrDeletion.ALOG).

11 At the command line prompt enter:

tail -f <job Id>.err

- **<job Id>.err** refers to the data processing log file to be reviewed.

12 Observe the log file to determine whether entries are being made in the file.

- If messages are being entered in the log file, there is probably no resource lock.

13 To quit tailing the log in the terminal window enter:

Ctrl-C

- A command line prompt is displayed in the terminal window.

14 Ensure that it is possible to connect to the necessary hosts and servers.
- For detailed instructions refer to the **Check Connections To Hosts/Servers** procedure (Section 13.7.1.1).

15 If no there is no database deadlock or resource lock and the Data Processing Subsystem servers (especially Deletion Server and Job Management Server) are up, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 13.7-15. Check for Resource Locks in the PDPS Database - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Log in to the PDPS database	Use procedure in Section 13.7.2.2
2	**Select * from DpPrResourceLock**	**enter text, press Enter**
3	**go**	**enter text, press Enter**
4	Analyze the results of the request.	**read text**
5	**delete DpPrResourceLock** or **delete DpPrResourceLock where jobId like "<job Id>"** (as applicable)	**enter text, press Enter**
6	**go** (if applicable)	**enter text, press Enter**
7	**quit**	**enter text, press Enter**
8	UNIX window (Queuing Server)	**single-click** or use procedure in Section 13.2.1
9	**cd /usr/ecs/<MODE>/CUSTOM/logs**	**enter text, press Enter**
10	**tail -f <job Id>.err**	**enter text, press Enter**
11	Observe the log file (Are entries are being made in the log file?)	**read text**
12	**Ctrl-C**	**enter text**
13	Check connections to hosts/servers	Use procedure in Section 13.7.1.1
14	Call the help desk and submit a trouble ticket (if applicable)	Use procedure in Chapter 8

13.7.4 Respond to DPR Deletion that Fails

When deleting DPRs or PRs from the Production Request Editor, both the Job Management Server and Deletion Server (both DPS servers) are called to clean up all PDPS database tables associated with the DPR or PR. It is possible that any failure or "hung" condition can be attributed to the Production Request Editor itself or to the Job Management Server or Deletion Server.

Table 13.7-16 presents (in a condensed format) the steps required to respond to DPR deletion that fails. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Access a terminal window logged in to the Queuing Server host.

- Examples of Queuing Server host names include **e0sps04**, **g0sps06**, and **l0sps03**.
- For detailed instructions refer to the **Log in to ECS Hosts** procedure (Section 13.2.1).

2 At the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- Change directory to the directory containing the Data Processing Subsystem log files (e.g., EcDpPrDeletionDebug.log).

3 If there is a Deletion Server Debug log, at the command line prompt enter:

pg <file name>

- **<file name>** refers to the Data Processing Subsystem log file to be reviewed (e.g., EcDpPrDeletionDebug.log).
- The first page of the log file is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **vi**, **view**) can be used to review the log file.

4 If there is a Deletion Server Debug log, review the log file for error messages.

- For example:

  Could not make database interface to DpPrFile
  Aborting: will report error to client.

- The message in the example indicates a compilation problem in Deletion Server so Deletion Server cannot communicate with the PDPS database.
  - When Deletion Server was compiled the RogueWave database (RWDB) libraries were not linked correctly.
  - The most likely cause is the fact that the Deletion Server Imakefile was changed so that the DPS make.options was included.
- To exit from **pg** at the **:** prompt enter:

  q

  - The command line prompt is displayed.

5 If an error message of the type shown in the previous step was present in the log, notify the Operations Controller/System Administrator of the problem.

Table 13.7-16. Respond to DPR Deletion that Fails - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	UNIX window (Queuing Server)	**single-click** or use procedure in Section 13.2.1
2	**cd /usr/ecs/<MODE>/CUSTOM/logs**	**enter text, press Enter**
3	**pg <file name>**	**enter text, press Enter**
4	Review the log file	**read text**
5	Notify the Operations Controller/System Administrator of the problem (if applicable)	**contact Operations Controller**

13.7.5 Handle a DPR Scheduling Failure

There are three principal sources of information concerning failures to activate DPRs from the Planning Workbench:

- The Planning Workbench GUI itself may provide an error message.
  - The message would indicate either success or failure.
  - It would not indicate why the request failed.
- The Planning Workbench Message Handler (the window that comes up before the Planning Workbench and to which the Workbench logs various messages).
  - Any error encountered should be logged to the Message Handler.
- The Planning Workbench logs (debug log and ALOG).
  - The logs are located in the /usr/ecs/<MODE>/CUSTOM/logs directory and begin with “EcPlWb” (e.g., EcPlWbDebug.log, EcPlWb.ALOG).
  - The logs can further refine the cause of the error.

If job activation fails from the Planning Workbench, one of the following errors is likely to be reported:

- "DPR Validation Failed"
- "information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}"

Accordingly, the following procedures may be involved in handling a DPR scheduling failure:

- Respond to a "DPR Validation Failed" Error.
- Respond to an "information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" Error.

13.7.5.1 Respond to a "DPR Validation Failed" Error

There are two conditions that may cause a "DPR Validation Failed" error to be reported:

- Performance data for the PGE are missing from the PDPS PlPgePerformance database table.

- Resource information for the PGE is missing from the PDPS PlResourceRequirement database table.

Consequently, it is necessary to check the appropriate PDPS database tables to determine whether the necessary PGE information is in the database.

Although the procedure for responding to a "DPR Validation Failed" error includes the use of isql commands, an acceptable alternative is to use a database browser to check the contents of the PlPerformance and PlPgePerformance tables.

Table 13.7-17 presents (in a condensed format) the steps required to respond to a "DPR Validation Failed" error. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Log in to the PDPS database.

- Database log-in is described in Steps 1 through 5 of the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).

2 At the **1>** prompt enter:

select * from PlPgePerformance

- Prepare a request to view the contents of the PlPgePerformance table.

3 At the **2>** prompt enter:

go

- The contents of the **PlPgePerformance** table are displayed.
- The listing includes data in the following columns:
  - **pgeId.**
  - **cpuTime.**
  - **pgeElapsedTime.**
  - **dprElapsedTime.**
  - **maxMemory.**
  - **faults.**
  - **swaps.**
  - **blockInputOperation.**
  - **blockOutputOperation.**
  - **runCpuTime.**
  - **runMaxMemory.**
  - **runPgeElapsed.**
  - **runDprElapsed.**
  - **runFaults.**
  - **runSwaps.**

- **runBlockInOperation.**
    - **runBlockOutOperation.**
    - **sharedMemory.**
    - **runSharedMemory.**
- For the PGE(s) that is (are) not schedulable, verify there are non-zero values for the entries in the table.
    - Zero may be appropriate for some columns but not for all.

4 At the **1>** prompt enter:

select * from PlResourceRequirement

- Prepare a request to view the contents of the PlResourceRequirement table.

5 At the **2>** prompt enter:

go

- The contents of the **PlResourceRequirement** table are displayed.
- The listing includes data in the following columns:
    - **sswId.**
    - **string.**
    - **numOfCPUs.**
    - **computer.**
    - **diskSpace.**
    - **topLevelShellName.**
    - **exeTarFileDiskSpace.**
    - **mcfName.**
    - **ramSize.**
    - **exeUntarFileDiskSpace.**
    - **exeTarUR.**
    - **pgeId.**
    - **toolkitArchitecture.**
    - **pgeCommands.**
- For the PGE(s) that is (are) not schedulable, verify that there are non-zero values for the entries in the table.
    - Zero may be appropriate for some columns but not for all.

6 To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.

7 If the entries in either the **PlPgePerformance** or **PlResourceRequirement** table are not appropriate, make a request to the SSI&T team to have the correct values entered.

8 Delete the affected DPR(s) as described in the **Delete a Data Processing Request** procedure (Section 13.2.8).

9 When the SSIT team has completed updating the PGE information in the PDPS database, create a new Production Request to replace the affected DPR(s) as described in the **Create a New Production Request Using the Production Request Editor GUI** procedure (Section 13.2.3).

10 Create and activate a new Production Plan (to send the replacement DPR(s) to data processing) as described in the **Create a New Production Plan** procedure (Section 13.4.2).

Table 13.7-17. Respond to a "DPR Validation Failed" Error - Quick-Step Procedures

Step	What to Enter or Select	Action to Take
1	Log in to the PDPS database	Use procedure in Section 13.7.2.2
2	**select * from PlPgePerformance**	**enter text, press Enter**
3	**go**	**enter text, press Enter**
4	**select * from PlResourceRequirement**	**enter text, press Enter**
5	**go**	**enter text, press Enter**
6	**quit**	**enter text, press Enter**
7	Ask the SSI&T team to have the correct values entered in the database tables (if applicable)	**contact SSI&T team**
8	Delete the affected DPR(s)	Use procedure in Section 13.2.8
9	Create a new Production Request to replace the affected DPR(s) when the SSIT team has completed updating the PGE information in the PDPS database	Use procedure in Section 13.2.3
10	Create and activate a new Production Plan (to send the replacement DPR(s) to data processing)	Use procedure in Section 13.4.2

13.7.5.2 Respond to an "information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" Error

When an "information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" error is reported on the Message Handler, one of two problems is likely:

- A predecessor DPR exited in a "bad" state.
  - The unschedulable DPR depends on the predecessor DPR for input data.
  - The predecessor DPR is in either a completed or an error status, or was not included in the activated plan.

- The current DPR has a "QA_FAILURE."
  - A metadata check or query for the DPR failed so the DPR cannot be scheduled.

Consequently, it is necessary to check the appropriate PDPS database tables and possibly the Planning Workbench debug log to localize the problem.

Although the procedure for responding to an "information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" error includes the use of isql commands, an acceptable alternative is to use a database browser to check the contents of the appropriate tables.

Table 13.7-18 presents (in a condensed format) the steps required to respond to a "DPR Validation Failed" error. If you are already familiar with the procedures, you may prefer to use the quick-step table. If you are new to the system, or have not performed this task recently, you should use the following detailed procedures:

1 Log in to the PDPS database.

- Database log-in is described in Steps 1 through 5 of the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).

2 At the **1>** prompt enter:

select * from PlDataProcessingRequest

- Prepare a request to view the contents of the PlDataProcessingRequest table.
- If there is data on many DPRs in the PlDataProcessingRequest table, it may be advisable to limit the search, for example:

  1> select * from PlDataProcessingRequest where dprId like "AM1Eph#30002282200TS2"

  - In the example **AM1Eph#30002282200TS2** is the predecessor DPR being checked.

3 At the **2>** prompt enter:

go

- The contents of the **PlDataProcessingRequest** table are displayed.
- The listing includes data in the following columns:
  - **dprId.**
  - **productionRequestId.**
  - **priority.**
  - **predictedStart.**
  - **actualStart.**
  - **completionState.**
  - **pgeId.**
  - **baselineTime.**

- **name.**
- **dataStartTime.**
- **dataStopTime.**
- **sswId.**
- **dprCollectionId.**
- **alarmTime.**
- **lateMessagesSent.**
- **autosysId.**
- **tileId.**
- **timeStamp.**

- For the predecessor DPR to the unschedulable DPR, verify that the **completionState** is **SUCCESS**.

4 To exit from **isql** at the **1>** prompt enter:

quit

- The connection with the database is discontinued.

5 If the **completionState** for the predecessor DPR to the unschedulable DPR is **not SUCCESS**, perform Steps 6 through 8; otherwise, go to Step 9.

- For example, if **completionState** is **FAILEDPGE** or **CANCELED**, perform Steps 6 through 8.

6 Delete the unschedulable DPR(s) and its predecessor(s) as described in the **Delete a Data Processing Request** procedure (Section 13.2.8).

7 Create new Production Request(s) to replace the affected DPR(s) as described in the **Create a New Production Request Using the Production Request Editor GUI** procedure (Section 13.2.3).

8 Create and activate a new Production Plan (to send the replacement DPR(s) to data processing) as described in the **Create a New Production Plan** procedure (Section 13.4.2).

- End of procedure.

9 If the **completionState** for the predecessor DPR to the unschedulable DPR is **SUCCESS**, at the command line prompt enter:

cd /usr/ecs/<MODE>/CUSTOM/logs

- For**<MODE>** type the desired operating mode.
- Change directory to the directory containing the appropriate log files (e.g., EcPlWbDebug.log).

10 At the command line prompt enter:

pg <file name>

- **<file name>** refers to the planning log file to be reviewed (e.g., EcPlWbDebug.log).
- The first page of the log file is displayed.
- Although this procedure has been written for the **pg** command, any UNIX editor or visualizing command (e.g., **more**, **vi**, **view**) can be used to review the log file.

11 Review the log file to determine whether the log contains an error message (concerning the unschedulable DPR) that indicates why the DPR generation failed.

- Look for an entry for the unschedulable DPR that contains either “QA_FAILURE” or “INVALID.”
  - "QA_FAILURE" indicates a problem with a metadata check/query so the DPR cannot be scheduled.
  - “INVALID” indicates that DPR validation failed.
- To exit from **pg** at the **:** prompt enter:

  q

  - The command line prompt is displayed.

12 If the log contains an error message that includes the term "INVALID", perform the **Respond to a "DPR Validation Failed" Error** procedure (Section 13.7.5.1).

13 If the log contains a "QA_FAILURE" error message, log in to the PDPS database.

- Database log-in is described in Steps 1 through 5 of the **Use ISQL to Check Database Tables** procedure (Section 13.7.2.2).

14 At the **1>** prompt enter:

select * from PlDprData

- Prepare a request to view the contents of the PlDprData table.
- If there is data on many DPRs in the PlDprData table, it may be advisable to limit the search, for example:

  1> select * from PlDprData where dprId like "ACT#syn1#004130123TS2"

  - In the example **ACT#syn1#004130123TS2** is the predecessor DPR being checked.

15 At the **2>** prompt enter:

go

- The contents of the **PlDprData** table are displayed.
- The listing includes data in the following columns:
  - **dprId.**
  - **granuleId.**

- **logicalId.**
- **primaryType.**
- **accepted.**
- **theOrder.**
- **type.**
- **temporalFlag.**
- **timeWait.**
- **ioFlag.**
- **timerExp.**
- **timerStart.**
- **numNeeded.**
- **waitForFlag.**
- **linkId.**
- **minGranReq.**

- Note that for each DPR there are separate entries for the different data types.
- For all entries for the unschedulable DPR, verify that the values in the **accepted** column are **not** equal to 2.
  - A value of 2 in the **accepted** column indicates that the metadata check/query for the corresponding input (**granuleId** column) failed.

16 At the **1>** prompt enter:

select * from PlEsdtParam

- Prepare a request to view the contents of the PlEsdtParam table.
- If there is data on many DPRs in the PlEsdtParam table, it may be advisable to limit the search, for example:

  1> select * from PlEsdtParam where dataTypeId like "AST_L1B#001"

  - In the example **AST_L1B#001** is the data type being checked.

17 At the **2>** prompt enter:

go

- The contents of the **PlEsdtParam** table are displayed.
- The listing includes data in the following columns:
  - **dataTypeId.**
  - **paramName.**
  - **paramType.**
  - **containerName.**
  - **secondParm.**
  - **secondValue.**
- Verify that there are entries for the appropriate **dataTypeId** in the PlEsdtParam table.

18 If no entry for the unschedulable DPR has a value equal to 2 in the **accepted** column, at the **1>** prompt enter:

select * from PlEsdtParmValues

- Prepare a request to view the contents of the PlEsdtParmValues table.

19 At the **2>** prompt enter:

go

- The contents of the **PlEsdtParmValues** table are displayed.
- The listing includes data in the following columns:
  - **esdtParmName.**
  - **esdtParmVal.**
  - **granuleId.**
  - **secondParm.**
  - **secondValue.**
- Verify that there are entries for the appropriate data type (in the **granuleId** column) in the PlEsdtParmValues table.

20 If there are entries for the appropriate data type in the **PlEsdtParam** and **PlEsdtParmValues** tables, at the **1>** prompt enter:

select * from PlMetadataChecks

- Prepare a request to view the contents of the PlMetadataChecks table.
- If there is data on many data types in the PlMetadataChecks table, it may be advisable to limit the search, for example:

  1> select * from PlMetadataChecks where dataTypeId like "AST_L1B#001"

  - In the example **AST_L1B#001** is the data type being checked.

21 At the **2>** prompt enter:

go

- The contents of the **PlMetadataChecks** table are displayed.
- The listing includes data in the following columns:
  - **pgeId.**
  - **dataTypeId.**
  - **paraName.**
  - **logicalId.**
  - **type.**
  - **operator.**
  - **value.**
  - **ioFlag.**
  - **queryFlag.**

- **queryType.**
- **secondParm.**
- **secondValue.**

- Verify that the entries for the appropriate data type in the PlEsdtParam and PlEsdtParmValues tables match the expected values in the PlMetadataChecks table.

22 If an entry for the unschedulable DPR has a value equal to 2 in the **accepted** column of the **PlDprData** table (Step 15), make a record of the fact that the PGE cannot be run on the specified input data.

- The metadata check/query for the corresponding input failed.
- It may be possible to either reingest or reprocess the input data that failed the metadata check or query.
  - Steps 17 through 21 can give indications as to which applies, based on the data type(s) involved.

23 If no problem is identified in the preceding steps, call the help desk and submit a trouble ticket in accordance with site Problem Management policy.

Table 13.7-18. Respond to an "information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" Error - Quick-Step Procedures (1 of 2)

Step	What to Enter or Select	Action to Take
1	Log in to the PDPS database	Use procedure in Section 13.7.2.2
2	**Select * from PlDataProcessingRequest**	**enter text, press Enter**
3	**go**	**enter text, press Enter**
4	**quit**	**enter text, press Enter**
5	Go to Step 9 if the **completionState** for the predecessor DPR to the unschedulable DPR is "SUCCESS"	
6	Delete the unschedulable DPR(s) and its predecessor(s)	Use procedure in Section 13.2.8
7	Create new Production Request(s) to replace the affected DPR(s)	Use procedure in Section 13.2.3
8	Create and activate a new Production Plan [end of procedure]	Use procedure in Section 13.4.2
9	**cd /usr/ecs/<MODE>/CUSTOM/logs** (if applicable)	**enter text, press Enter**
10	**pg <file name>**	**enter text, press Enter**
11	Review the log file	**read text**
12	Respond to the "DPR Validation Failed" error (if applicable)	Use procedure in Section 13.7.5.1
13	Log in to the PDPS database (if applicable)	Use procedure in Section 13.7.2.2
14	**select * from PlDprData**	**enter text, press Enter**

Table 13.7-18. Respond to an "information (INFO) Production Request {Production Request Id} has unschedulable DPR {DPR Id}" Error - Quick-Step Procedures (2 of 2)

Step	What to Enter or Select	Action to Take
15	**go**	**enter text, press Enter**
16	**select * from PlEsdtParam**	**enter text, press Enter**
17	**go**	**enter text, press Enter**
18	**select * from PlEsdtParmValues** (if applicable)	**enter text, press Enter**
19	**go**	**enter text, press Enter**
20	**select * from PlMetadataChecks** (if applicable)	**enter text, press Enter**
21	**go**	**enter text, press Enter**
22	Make a record of the fact that the PGE cannot be run on the specified input data (if applicable)	**enter text**
23	Call the help desk and submit a trouble ticket in accordance with site Problem Management policy (if applicable)	Use procedure in Chapter 8